

BSD

FOR NOVICE AND ADVANCED USERS

Adobe® PDF
Magazine Version

THE WONDERS OF BLENDER

INSIDE

RAMBLINGS FROM THE ROGUE ADMIN

RUN YOUR PHONE SYSTEM ON OPENBSD

A QUICK LOOK AT THE UPCOMING PC-BSD 9

DRUPAL ON FREEBSD – PART 4

HOW TO SETUP OPENBSD ON THE EMBEDDED ALIX CARD

USING FREEBSD TO AUTHENTICATE USERS WITH OPENLDAP AND FREERADIUS

SETTING UP GIT AND MERCURIAL SERVERS

USEFUL OPENBSD TOOLS

VOL.4 NO.3
ISSUE 03/2011 (20)
1898-9144



800-820-BSDI
<http://www.iXsystems.com>
Enterprise Servers for Open Source



✓ Increased Performance ✓ Impressive Energy Savings

10 Gig On Board

On-Board 10 Gigabit Ethernet Adapters leave your existing PCI-E slots available for other expansion devices.



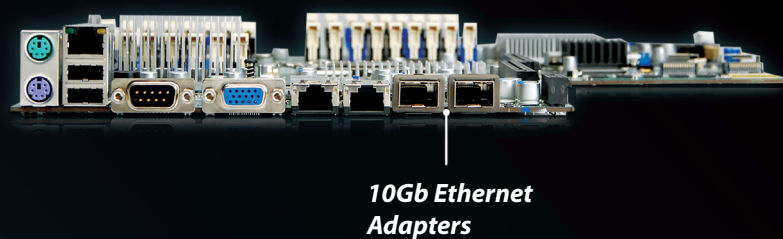
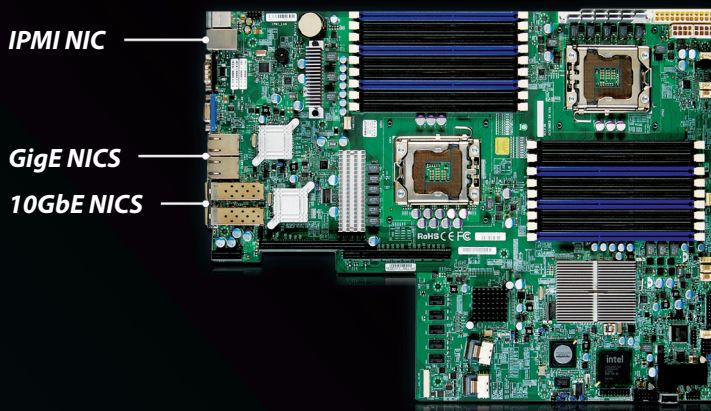
30% cost savings/port over equivalent Dual-Port
10 GB PCI Express add-on card solution



Blazing Fast, Embedded 10Gb Ethernet

10G Rackmount Servers in the iX-Neutron server line feature the Intel® Xeon® Processor 5600/5500 Series, and come with 10GbE networking integrated onto the motherboard. This eliminates the need to purchase an additional expansion card, and leaves the existing PCI-E slots available for other expansion devices, such as RAID controllers, video cards, and SAS controllers.

For more information on the iX-1204-10G, or to request a quote, visit: <http://www.iXsystems.com/neutron>



KEY FEATURES:

- Supports Dual 64-Bit Six-Core, Quad-Core or Dual-Core, Intel® Xeon® Processor 5600/5500 Series
- 1U Form Factor with 4 Hot-Swap SAS/SATA 3.5" Drive Bays
- Intel® 5520 chipset with QuickPath Interconnect (QPI)
- Up to 192GB DDR3 1333/1066/800 SDRAM ECC Registered Memory (18 DIMM Slots)
- 2 (x8) PCI-E 2.0 slots + 1 (x4) PCI-E 2.0 (in x8 slot -Low-Profile - 5.5" depth)
- Dual Port Intel® 82599EB 10 Gigabit SFP+ - Dual Port Intel® 82576 Gigabit Ethernet Controller
- Matrox G200eW Graphics
- Remote Management - IPMI 2.0 + IP-KVM with Dedicated LAN
- Slim DVD
- 700W/750W Redundant AC-DC 93%+ High-Efficiency Power Supply



Call iXsystems toll free or visit our website today!
1-855-GREP-4-IX | www.iXsystems.com

Intel, the Intel logo, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation in the U.S. and/or other countries.



Dear Readers!

Let me present you with March issue of BSD Magazine.

This month start will be something new, and I am sure everyone will agree it is very interesting. :) We have invited developers of all the biggest BSD projects to write articles related to their work, and present it to our readers.

You can see the effect on following pages, where you will find great articles from Josh Paetzel, Kris Moore, Ian Darwin, and hot news DragonflyBSD new from Justin C. Sherrill.

After that we go into „How To's” – after a short break you will surely be excited to see another part of Drupal articles by Rob Sommerville, then learn about FreeRADIUS with Brivaldo Junior, followed by Guillaume Duale and James P. Howard, II and their tutorials.

In the final part of this issue Sufyan and Girish will present few interesting tools to us.

Enjoy your reading!

Thank you!

*Zbigniew Puchciński
Editor in Chief
zbigniew.puchcinski@software.com.pl*

MAGAZINE BSD

Editor in Chief:

Zbigniew Puchciński
zbigniew.puchcinski@software.com.pl

Contributing:

Josh Paetzel, Ian Darwin, Kris Moore, Rob Sommerville, Sufyan bin Uzayr, Guillaume Duale, Brivaldo Junior, James P. Howard, II, Girish Venkatachalam, Justin C. Sherrill

Proofreaders:

Corby Agid, Melanie Vonfange

Top Betatesters:

Simon Huang, Navin Seshadri

Special Thanks:

Denise Ebery, Matt Olander

Art Director:

Ireneusz Pogroszewski

DTP:

Ireneusz Pogroszewski

Senior Consultant/Publisher:

Paweł Marciniak pawel@software.com.pl

CEO:

Ewa Dudzic
ewa.dudzic@software.com.pl

Production Director:

Andrzej Kuca
andrzej.kuca@software.com.pl

Executive Ad Consultant:

Karolina Lesińska
karolina.lesińska@bsdmag.org

Advertising Sales:

Zbigniew Puchciński
zbigniew.puchcinski@software.com.pl

Publisher :

Software Press Sp. z o.o. SK
ul. Bokserka 1, 02-682 Warszawa
Poland
worldwide publishing
tel: 1 917 338 36 31
www.bsdmag.org

Software Press Sp z o.o. SK is looking for partners from all over the world. If you are interested in cooperation with us, please contact us via e-mail: editors@bsdmag.org

All trade marks presented in the magazine were used only for informative purposes. All rights to trade marks presented in the magazine are reserved by the companies which own them.

The editors use automatic DTP system **AQAPDS**

Mathematical formulas created by Design Science MathType™.

Get Started

08 Ramblings from the Rogue Admin

Josh Paetzel

FreeBSD is a rapidly evolving target, which can be a surprise to many people used to FreeBSD.

12 Run your Phone System on OpenBSD

Ian Darwin

Who says you can't run your telephone system on the most secure OS around? Not me, for sure: I run two Asterisk installations on OpenBSD.

14 A quick look at the upcoming PC-BSD 9

Kris Moore

Even though the release of PC-BSD 9.0 is still a little ways off in 2011, there has already been countless hours of work put into it, bringing many exciting new changes and features.

How To's

16 Drupal on FreeBSD – part 4

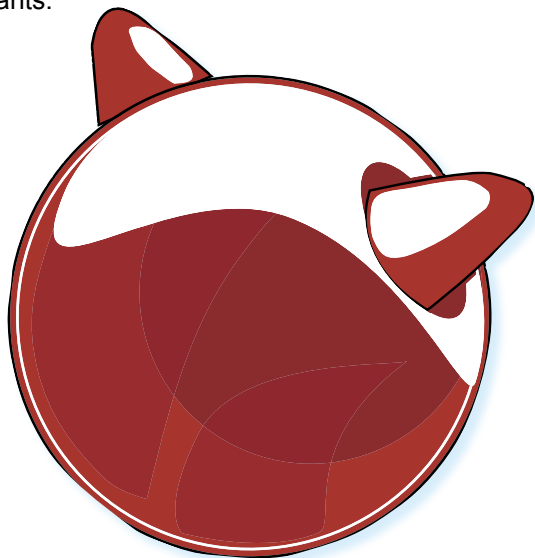
Rob Somerville

Continuing the series on the Drupal Content Management System, we will look at creating a basic time-slot booking system.

24 Using FreeBSD to authenticate users with OpenLDAP and FreeRADIUS

Brivaldo Junior

We introduce a WIFI authentication environment using 802.1X with a RADIUS server (FreeRADIUS), a central database (like OpenLDAP) to store user and password, and using MSCHAPv2 protocol to avoid third party supplicants.



30 How To Setup Openbsd On The Embedded Alix Card

Guillaume Duale

In this article you will learn how to setup a real operating system on an ALIX card. It's a mandatory step in the life of a System Administrator. With this guide you will survive to the hostile Internet ! Tremble...

34 Setting up Git and Mercurial Servers

James P. Howard, II

GitHub provides an excellent web-based interface to Git with extensive project management tools. Bitbucket provides an equally excellent web-based interface for Mercurial.

Tools

36 The Wonders Of Blender

Sufyan bin Uzayr

Blender is a powerful software, but can also be daunting, especially for BSD users, as the award-winning software isn't yet officially favored on BSD. Fear not! Let's explore this wonderful tool, starting with the user interface.

42 Useful OpenBSD Tools

Girish Venkatachalam

Generally speaking the UNIX world is famous for the rich repertoire of tools it gives and the way it integrates with the rest of the system.

Conferences

DrupalCon Chicago 2011

March 7-10
Chicago, USA



<http://chicago2011.drupal.org/>

AsiaBSDCon 2011

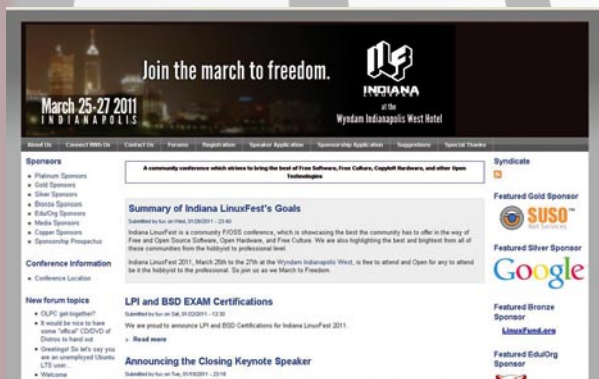
March 17-20
Tokyo, Japan



<http://2011.asiabsdcon.org/index.html>

Indiana LinuxFest 2011

March 25-27
Indianapolis, USA



<http://www.indianalinux.org/cms/>

Flourish! 2011 Open Source Conference

April 1-3
Chicago, USA



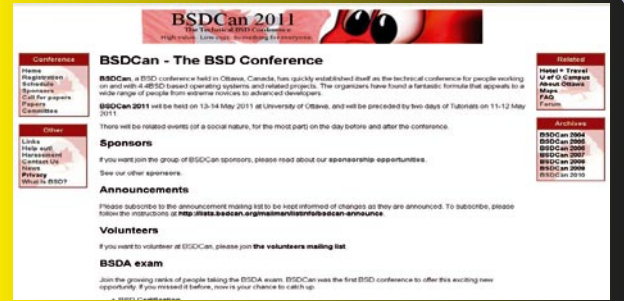
<http://www.flourishconf.com/2011/>

Conferences

BSDCan 2011

May 13-14
Ottawa, Canada

<http://www.bsdcan.org/2011/>



Open Source Business Conference

May 16-17
San Francisco, USA



<http://www.eiseverywhere.com/>

Ohio LinuxFest 2011

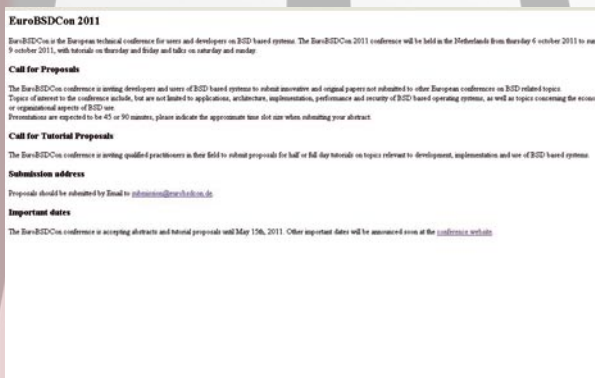
September 7-11
Columbus, Ohio, USA



<http://ohiolinux.org/>

EuroBSDCon 2011

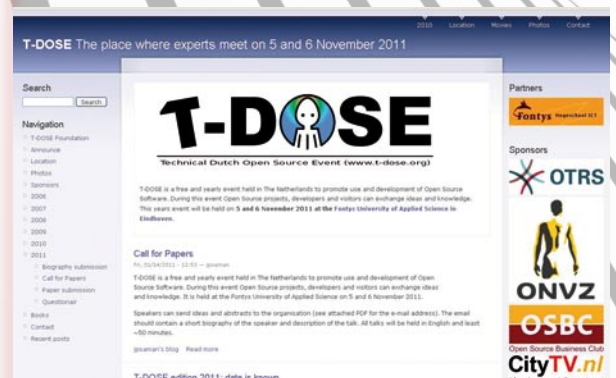
October 6-9
Netherlands



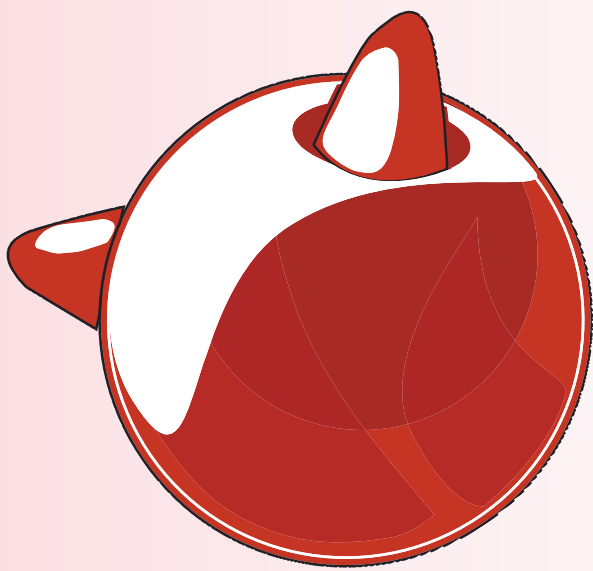
<http://2011.eurobsdcon.org/CfP.html>

T-DOSE 2011

November 5-6
Eindhoven, Netherlands



<http://www.t-dose.org/>



Ramblings from the Rogue Admin

FreeBSD is a rapidly evolving target, which can be a surprise to many people used to FreeBSD.

For a very long time we were attracted to BSD UNIX because of its stability and conservatism, and FreeBSD certainly continued that tradition for years. Due to that conservatism, there was a ton of knowledge transfer between FreeBSD versions, what you knew about FreeBSD 3.3 applied largely wholesale to FreeBSD 4.1.

For better or worse, those days are behind us. To make matters worse, in spite of the reality of the changes, the mindshare still persists. It's hard for someone has been using FreeBSD a long time to shake their belief that the stuff they knew about FreeBSD 4.x probably applies to FreeBSD 8.x or HEAD.

The reality is, all bets may be off! That's a good thing, as long as you recognize it.

So, in the spirit of splashing cold water at commonly held beliefs, here are a few things worth looking at. When reading these please keep in mind that individual results can vary greatly from overall trends, and a given set of concrete examples may appear to differ radically from trends.

Widespread belief

FreeBSD has good default tunings, and is designed to perform adequately under a wide range of use cases, but to unlock its full performance for a given use case it requires a good deal of tuning.

Response

A lot of work has gone into the auto-tuning capabilities of FreeBSD in the 8.x releases and HEAD. In many cases statically configuring the OS can hinder its ability to auto-tune and can actually hurt performance. When embarking on a journey to tune modern FreeBSD care needs to be taken to measure your existing performance

to ensure that you aren't hurting more than you're helping.

Widespread belief

Only releases are production grade. One should avoid running development branches in production.

Response

There is certainly much hard-won wisdom in this attitude, but there's another side to the coin. FreeBSD has a very conservative userbase, and oftentimes only production releases are subjected to production workloads, which means that 8.2 doesn't see as much use until after it's released. As it gets more and more use bugs are noticed and fixed in the `RELENG_8` branch, that will be the basis of 8.3. Oftentimes `STABLE` shortly after a release is a fine candidate for improvements that aren't going to become Errata Notices for the previous release, and that you'd otherwise have to wait for the next release to get.

Widespread belief

FreeBSD has excellent documentation.

Response

FreeBSD has excellent documentation! Unfortunately it's not always current documentation. The rate of change in FreeBSD can outstrip the efforts to keep the documentation up to date. This becomes magnified with many people tossing their workflows and experiences up on the web. While their procedure might have been fine on FreeBSD 7.2, there's no telling how valid it is on FreeBSD 8.1. Beware google. Beware man pages that haven't been updated since the source was touched, especially when dealing with device drivers.

Widespread belief

ZFS on FreeBSD is new and relatively untested, definitely not production ready.

Response

ZFS on FreeBSD is definitely newer than UFS. It's also a port from another operating system. While it can have it's share of issues and hiccups it is in many cases ready for your production workloads. You wouldn't put a new RAID controller into production without spending some time with it seeing how it works. You'd try out drive replacement, monitoring it with CLI tools, checking performance, and ZFS is no different. Try it on a backup server, put it in a VM, spend some time learning it's features, getting familiar with it.

You might find yourself pleasantly surprised. In many cases you'll come to love features it provides to the point where you don't know how you did without them.

FreeBSD is a powerful and flexible operating system. Search out people who are using it extensively in production. In some cases you'll find they are using undocumented tips and tricks that can make your life easier. Sometimes their efforts aren't pushed into the FreeBSD documentation project simply because of time constraints.

For those of you gracious enough to read to the very end, allow me to drop just a couple of small hints that you may or may not find useful.

Consider making ethernet interfaces a part of a lagg device, even if you are using a single device at the moment. You never know when you might need to reconfigure your network, and moving from a single switch to redundant switches is just one example where having a device in a lagg means such a migration in network topology can account for no loss in network connectivity. Failover lags are particularly useful, as they require no switch configuration, and can be created with a single device initially.

mps is a device driver for the new LSI 6 mbps HBA controllers, which can be found rebranded in new models from many OEMs such as Dell, HP, and IBM. The driver didn't make it into 8.2-RELEASE, but it is in RELENG_8 now, so if you have a new system where your disks aren't detected, and you suspect you might have a new LSI controller in it, check out STABLE, or feel free to ping the mailing lists, as the driver is available as a ko and will work on 8.2 or 8.1.

JOSH PAETZEL

A 37 year old advocate, user and developer of BSD UNIX based systems. he resides in Minneapolis, Minnesota, USA where he hacks on FreeBSD and PC-BSD, both as a volunteer and as part of his full time work as the Director of IT at iXsystems.

Visit our website

You will find here:

- ➡ **materials for articles-
listings, additional
documentation, tools**
- ➡ **the most interesting
articles to download**
- ➡ **current information
on the upcoming
issue**

www.bsdmag.org

DragonFly News

Multiprocessor progress

The real key to multiprocessor improvement in DragonFly is the same as with other BSDs: removing the giant kernel lock. DragonFly has been moving away from blocking mechanisms, and implementing a token system. All the global tokens in DragonFly have now been updated to run without that giant lock.

A number of other systems were updated. The `tmpfs(5)` filesystem now runs without multiprocessor locks. `fork()` and `exit()` are also now multiprocessor safe.

The giant lock is no longer the largest source of contention in DragonFly. This has made a significant difference in speed and interactivity.

Pkgsrc progress

Rumko, one of the DragonFly developers, has been tackling broken packages with great success. The most recent quarterly release of pkgsrc, 2010Q4, is averaging only around 350 packages out of approximately 10,500 failing to build, which is an excellent result, especially compared to previous quarterly release.

Binary packages for pkgsrc-2010Q4 have been built and uploaded for general use. They can be retrieved using `pkg_radd` on a DragonFly system, or one of the many binary package installers for pkgsrc, like `pkgin`. (`pkgin` received an update to 0.4 in this time period, though that update is not yet part of the binary packages for DragonFly.) Packages were built for DragonFly 2.8 and DragonFly 2.9, on the i386 and `x86_64` platforms.

Package bulk build results are available at <http://avalon.dragonflybsd.org/reports/> and often sent to the pkgsrc-bulk@netbsd.org mailing list.

pkgsrc-current now understands the `bin-install` target for DragonFly. Pkgsrc will attempt to download binary packages without any further configuration, and drop

back to building from source if the download fails. This was only an automatic operation for NetBSD, previously.

Hammer updates

Hammer is the default file system in DragonFly. Hammer is designed to provide fine-grained history and snapshots, networked mirroring, and instant crash recovery. It works well on multiple huge drives, and across slow links for immediate streaming backup.

Hammer reached version 5 recently, which means it has support for data deduplication. Hammer deduplication was originally run as a batch process similar to other disk cleanup options. It also now works *live*, also called the *fast cp* option. Hammer will look at the data being copied and if it's duplicated on disk, just update the index of what information is referenced instead of actually moving around additional copies of data. This leads to a huge speed gain for even common tasks, like `cp`.

Deduplication is available in the daily snapshots of DragonFly. The current release version, DragonFly 2.8, has Hammer support but no deduplication.

dragonflybsd.org changes

dragonflybsd.org is hosted at the home of Matthew Dillon. The site has been relatively low on bandwidth for some time, making it take a relatively long time to complete a new download of DragonFly source. This hasn't been much of a disruption, but then again, nobody ever complains about their network connection running too fast.

Matthew added a AT&T U-Verse connection and ended up rewriting the `bridge(4)` driver to accomodate bonding all his network connections together. The `link0` feature enables transparent bridging, where the source interface's MAC address is carried through to the other side of the bridge. The `link1` feature features automatic failover between all the interfaces attached to that bridge.



DragonFlyBSD

The end result is much better bandwidth for dragonflybsd.org, plus a variety of possible network paths.

Scheduler updates

DragonFly has a scheduler framework, where multiple schedulers can be placed in the system and switched at startup. The default *bsd4* scheduler has gone through a great deal of refinement, and in February received further updates with repeated testing using parallel makes and *blogbench*.

Google Summer of Code

DragonFly is hoping to participate in Google Summer of Code for a fourth year. The application for DragonFly to be a mentoring organization is already in as of this writing. There's at least 8 potential mentors lined up, so this could be a very successful year in terms of total volume.

Last year had 3 projects, all of which were successful. We've been involved long enough that some of the possible mentors actually started as students in Google Summer of Code.

A list of potential projects for Summer of Code and DragonFly has been posted at the DragonFly website. The site is a wiki, so anyone is welcome to add ideas, or contribute to the ones already there.

Google Code-In

DragonFly was the one BSD project participating in Google Code-In 2010. Google Code-In is a similar project to Google's Summer of Code, but designed to have smaller tasks in larger quantity, for people 12 to 18 years of age.

DragonFly mentors came up with a large list of tasks, many of which are documented on the DragonFly website. Roughly a half of the tasks were devoted to documentation, which were not attempted by many students. Documentation work as a paid activity has come up in multiple years for Google Summer of Code, but it did not prove to be popular now that the ability to actually write documentation is there. The tasks that were completed included:

- pkgsrc package fixes
- conversion of various systems from *zmalloc* to *objcache*
- a *devattr* tool
- documentation strings for many *sysctl* areas

- EXAMPLES sections for many manpages
- ... and many other tasks

DragonFly had 72 tasks completed, some of which we had originally thought would be beyond a teenager's ability. Most, if not all, of the work from Google Code-In has been committed to DragonFly by Samuel Greear.

Other recent updates

Jan Lentfer has updated the version of *pf* in DragonFly to the 4.4 equivalent, keeping some DragonFly-specific updates like *fairq*, designed to keep network connections responsive while at or near capacity.

Peter Avalos updated a variety of basic utilities like *sh*, *kill*, *test*, and *printf*, using recent changes from FreeBSD. OpenSSL was updated to 1.0.0.d. Peter also updated *file* to version 5.05. It's strange to think of *file* as a separate utility from BSD, since it's been included in the base system for almost 4 decades.

Sepherosa Ziehau has been steadily updating interrupt support on DragonFly.

More modern motherboards will be supported by these changes.

The *ps* utility has a *-R* option, which sorts processes by their parent/child relationship, and indents lines to make the relationship clear. Minor as this may seem, it's something that would have been useful two decades ago.

Tim Bisson and Pratyush Kshirsagar have been working on drivers for DragonFly to use when under emulation. These *virtio* drivers have made some progress, though some of the original FreeBSD code turns out to not be under the BSD license. The current *virtio* drivers for DragonFly no longer have any of that code. There's a good chance that these same type of drivers will show up as a Google Summer of Code project, too.

Coming soon: Work is underway to set *gcc 4.4* as the system default compiler, by Sascha Wildner. It's already available as an option, as is building with *clang* or *pcc*.

JUSTIN C. SHERRILL



Run your Phone System on OpenBSD

Who says you can't run your telephone system on the most secure OS around? Not me, for sure: I run two Asterisk installations on OpenBSD.

What you will learn...

- basic concepts of Asterisk open-source VOIP package
- example uses of Asterisk
- using the provided packages to get started with Asterisk on OpenBSD

What you should know...

- Basics of running an OpenBSD system
 - Starting services with OpenBSD's new `/etc/rc.d/`
 - Editing configuration files
 - Configuring pf firewall
-

Asterisk is one of the leading open-source telephony systems. It was originally written by Mark Spencer; Mark went on to found Digium to sell hardware to support the open-source model; this combination has done very well for Digium and the open source VOIP world.

Open source VOIP systems are much more malleable, flexible, configurable, than your average *black box* commercial telephone system, and there are no royalties, hidden per-line or per-mailbox fees, and it runs on a regular computer system. Spencer's founding dictum was something like *Telephony? Voice? It's only data...*

Interestingly, Digium now sells an *appliance* PBX system based on digium, but the vast majority of Asterisk systems (<http://asterisk.org/>) run on Linux and BSD. Check out <http://digium.com/> as well as <http://voip-info.org/> for more background.

There is one catch for OpenBSD – at present, you can't use the commonly-used Digium or Sangoma cards for line termination. Your choices are to use an Analog-SIP converter, or to have lines terminated at a Voice Over IP house. I use each of these solutions at one of my two sites. If you really feel you need to bring 4 or 8 or 32 analog *Tel/co* lines directly into your box, then you might have to run FreeBSD. The drivers for the Digium cards are covered by the GPL, which means they cannot be incorporated into the base system of OpenBSD

(that's just OpenBSD's policy). There is a project which has made them work on FreeBSD via kernel module loading (*modload*) and one could probably adapt this to OpenBSD, but I don't think anybody's had the time and inclination to do so.

So in the meantime, we have two approaches, VOIP and analog, to get connections to the phone system.

In my work/voicemail system, I use the VOIP service provided by local provider Unlimitel (<http://www.unlimitel.ca/>). Unlimitel has a great reputation for service, and their leader, Stephen Monette, has been supportive of the local Asterisk User's Group (<http://taug.ca/>).

I rent some number of *DID* lines. These are terminated from the phone company by Unlimitel (they terminate thousands of lines), and feed them through to customers' VOIP systems over the Internet. The term *DID* used to stand for Direct Inbound Dialing, but nowadays in this context is used just to mean a VOIP line that you rent from a VOIP supplier. Each line is connected into my Asterisk system, so that when somebody dials my public phone number, the call is routed through to my Asterisk server; when I dial out, my Asterisk server initiates a call to the VOIP supplier, and they call out. Each DID can handle several concurrent conversations, so a small office can often get by with one DID for both inbound and outbound.

Asterisk supports several protocols; the most common are SIP and IAX. SIP, the Session Initiation Protocol, is commonly used by VOIP phones, and is supported by most *software phones* or softphones.

IAX2 (pronounced *eeks two*) is the current version of the Inter-Asterisk Exchange protocol, intended for use between Asterisk instances, or from Asterisk to some other phone system or VOIP provider. IAX2 is also supported by a few VOIP telephones and softphones, but is most widely used between phone systems. I prefer IAX2 mainly because SIP is more widely known and thus more widely subjected to break-in attempts from crackers and script kiddies.

Needless to say in this setup I have PF set to allow outgoing IAX calls, and to allow incoming IAX, but only from the provider's static IP address. Since I don't use SIP, I have been immune to most of the more common attacks against Asterisk servers – there have been a few over the years.

In my home system, I use an *analog terminal adapter* (ATA) to connect my home line to the analog telephone network. I live way out in the country where internet access generally sucks (no DSL, not even ISDN!). So using a VOIP provider here is not an option. I have Asterisk running on OpenBSD talking to a Grandstream ATA, which in turn talks to the analog network. Incoming calls will ring through to my VOIP phone. I use a Polycom IP500 desk set, since I long ago configured its complex mess of XML files, and don't want to change.

My previous attempts to install VOIP phones in the rest of the house did not pass *the wife test*, alas. So the other phones in the house are plain analog phones, meaning I can't transfer calls from my VOIP line to the other phones, but in practice it works not badly. Here, everything is behind my firewall, so I don't need to allow either SIP or IAX2 in or out of my firewall.

The hardware this runs on is interesting – it's in a regular PC cabinet, but it runs on an Intel D201GLY Mini-ITX motherboard which is very low power – the whole system runs on a 12 Watt power supply.

And it runs in 64-bit mode (what OpenBSD calls *amd64*). It's plugged into a KVM so doesn't have a dedicated monitor to get left on and waste electricity.

Besides Asterisk itself, there are several related programs in OpenBSD's ports/packages systems. On the sound front, asterisk-sounds provides additional sound files for use in interactive voice response and related

systems. Asterisk-native-sounds provides some better-sounding versions of the *standard* voice files that come with Asterisk.

Asterisk-openbsd-moh provides the OpenBSD release songs for use as Asterisk music-on-hold programming. Royalty-free, as you'd expect! If you don't like these, you can download royalty-free music from a variety of places (I have used <http://freeplaymusic.com/> in the past).

There is also appkonference, a conferencing application for asterisk. Gsutil lets you dump/restore Grandstream device configurations – needless to say I keep my ATA configuration backed up!

laxmodem, which I have never even tried to use, claims to be a software FAX modem using an IAX channel. Astmanproxy is a proxy for the Asterisk *Manager* Interface – an administrative API. p5-asterisk offers some PERL modules to be used with Asterisk. On the client side, we have pjsui for SIP and iaxclient for IAX. Also, Ekiga can use SIP to talk to an Asterisk server.

There is also a package books/Asterisk-TFOT which installs the Creative-Commons-Licensed book *Asterisk: The Future Of Telephony*, which will tell you more about Asterisk and all the neat things it can do, as well as how to modify the sample configuration files that the Asterisk package installs.

So, apart from the lack of drivers for the analog cards, OpenBSD has good support for Asterisk, and makes a good security-friendly platform to build and run telephony applications.

If telephony is one of your things, why not give it a try? Just set your `PKG_PATH` to a local mirror and do

```
$ sudo pkg_add -v asterisk \
    asterisk-openbsd-moh Asterisk-TFOT
```

and read the PDF file `/usr/local/share/doc/asterisk/AsteriskTFOT-2.0.pdf`.

Then start in on the configuration files...

IAN DARWIN

*Ian Darwin is an OpenBSD committer who lives in the country well north of Toronto, Canada. He runs *NIX on just about all his computers; he once said that his only Windows looked out over the hillsides where he lives.*



A quick look at the upcoming PC-BSD 9

Even though the release of PC-BSD 9.0 is still a little ways off in 2011, there has already been countless hours of work put into it, bringing many exciting new changes and features.

PC-BSD

Probably the biggest and most noticeable change will be the ability to select from a variety of desktops/window managers. Historically PC-BSD has only offered KDE, starting with version 3, and later version 4 as a users main desktop. While KDE still offers a very complete desktop environment, there are a large number of users who prefer to use an alternative on their system. This is often for a variety of reasons, such as size, speed, design, or just personal preference. In order to provide a more satisfactory desktop experience to a larger audience, starting in version 9.0, users will be provided with a easy-to-use desktop selection screen, which will allow PC-BSD to be customized with the desktop packages of the users choice.

Currently some of the desktops being offered include KDE, GNOME, XFCE and LXDE. In addition to these desktops, some common packages are also offered for installation, such as NVIDIA drivers, HPLIP and MythTV. After an installation, sometimes a user may need to add or remove various packages and PC-BSD 9 provides a mechanism for this as well. By running the included *System Manager* tool, a user can quickly change the installed *meta-pkgs* again to their preference, by inserting the original DVD/USB media, or by installing from the Internet.

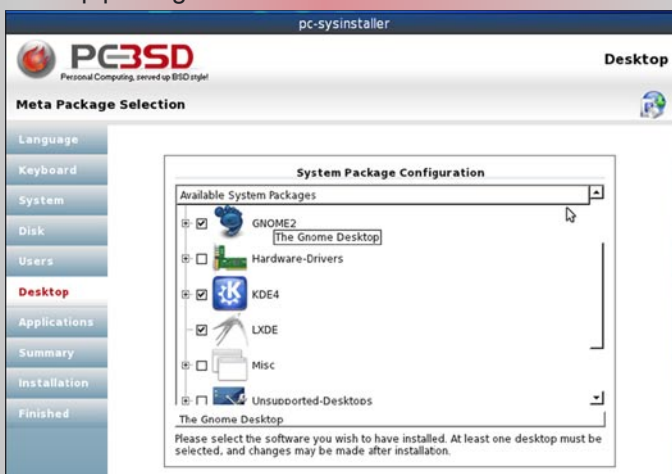


Figure 1. Desktop selection

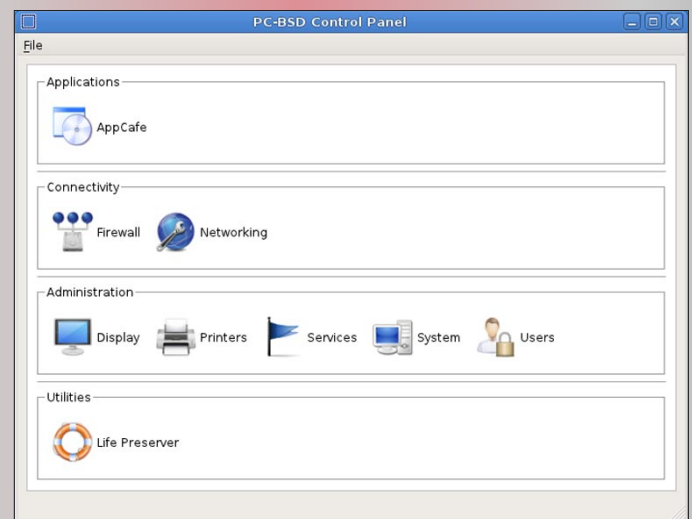


Figure 2. Control panel

In order to accommodate this large shift from a single desktop environment, almost all of the PC-BSD management tools have had to either be fixed, or in some cases replaced entirely. Since most of the desktops have a variety of different configuration managers, or none at all, it was decided to create our own *PC-BSD control panel*, which could provide a consistent interface for common configuration tasks. From this new control panel, a user can easily perform tasks such as setting up networking, add/removing users, controlling the firewall, browsing & installing software (PBIs) and more.

This brings us to the last major change to PC-BSD 9, the PBI package management system. In previous releases of PC-BSD, the PBI system had been developed with QT/KDE and was tied into that particular desktop in many ways. However, with the possibility of a user not even having KDE installed on their system, this meant our PBI system would need to change as well. It was decided to re-implement the PBI format entirely as command-line applications, so that it would be agnostic to the particular desktop being used, as well as be able to function on traditional FreeBSD systems, which may not even have X11 installed.

Since the entire PBI format was going to be overhauled for 9, we have also taken the opportunity to enhance it with a number of new features. Since a PBI file includes all the required libraries/dependencies included within it, there is a potential for file duplication between applications. In order to reduce this from occurring, the revamped PBI format includes intelligent management of libraries, and is able to share identical copies between applications. We have also added other important features, such as repository management, digital signature verification, off-line repository browsing and more. All these features are available via a command-line interface for power-users, while a new GUI front-end provides users of previous PC-BSD versions with a familiar framework for management.

Even though PC-BSD 9 is still early in the development cycle, it has already undergone some dramatic changes, and is shaping up to be a large step forward for BSD on the desktop. Testers or curious users are welcome to follow the development of this release by watching our new blog: <http://blog.pcbsd.org>.

KRIS MOORE

Kris Moore is the founder and lead developer of PC-BSD. He lives with his wife and four children in East Tennessee (USA), and enjoys building custom PC's and gaming in his (limited) spare time. kris@pcbsd.org

If you wish to contribute to BSD magazine, share your knowledge and skills with other BSD users – do not hesitate – read the guidelines on our website and email us your idea for an article.

Join our team!



Become BSD magazine Author or Betatester

As a betatester you can decide on the contents and the form of our quarterly. It can be you who read the articles before everybody else and suggest the changes to the author.

Contact us:
editors@bsdmag.org
www.bsdmag.org

Drupal on FreeBSD

Part 4

Continuing the series on the Drupal Content Management System, we will look at creating a basic time-slot booking system.

What you will learn...

- How to expand Drupal with the calendar and trigger modules

What you should know...

- Basic BSD system admin skills and how to install / administer Drupal CMS (Parts 1, 2 & 3)

One of the great benefits of the Drupal CMS is that with the extensive collection of third party modules available, many application challenges can be addressed without resorting to writing code. In the situations where coding is required, Drupal provides an extensive API although this does come with the proviso *Do not hack core!* By modifying the core Drupal

files (which is tempting when a quick and dirty fix is required) the short term gains rarely outweigh long-term stability and best practice: if the code and the modification is not thoroughly documented, any updates at a later date may overwrite your changes. Worse still, if they your code is not included in the main Drupal tree as a patch there may be other unforeseen interoperability issues, and peer review is useful for identifying these *Gotcha's*. Best practice is therefore to either add discrete code via the PHP filter module, or use/write a module to suit and

Enabled	Name	Version	Description
✓	Calendar	8.x-2.4	Views plugin to display views containing dates as Calendars. Depends on: Views (enabled), Date API (enabled), Date Timezone (enabled) Required by: Calendar iCal (enabled), Calendar Multiday (disabled), Calendar Popup (enabled)
✓	Calendar iCal	8.x-2.4	Adds iCal functionality to Calendar views. Depends on: Views (enabled), Date API (enabled), Calendar (enabled), Date Timezone (enabled)
✓	Calendar Multiday	8.x-2.4	Use the calendar multiday display as an alternative to the standard display. (BETA version) Depends on: Calendar (enabled), Views (enabled), Date API (enabled), Date Timezone (enabled)
✓	Calendar Popup	8.x-2.4	Replaces the links to calendar items with a javascript popup that gracefully regresses if javascript is not enabled Depends on: Calendar (enabled), Views (enabled), Date API (enabled), Date Timezone (enabled)
✓	Date	8.x-2.7	Defines CCK date/time fields and widgets. Depends on: Content (enabled), Date API (enabled), Date Timezone (enabled) Required by: Date Tools (enabled)
✓	Date API	8.x-2.7	A Date API that can be used by other modules. Required by: Calendar (enabled), Calendar iCal (enabled), Date (enabled), Date Locale (disabled), Date PHP4 (disabled), Date Popup (enabled), Date Repeat API (enabled), Date Timezone (enabled), Calendar Multiday (disabled), Date Tools (enabled), Calendar Popup (enabled)
✓	Date Locale	8.x-2.7	Allows the site admin to configure multiple formats for date/time display to tailor dates for a specific locale or audience. Depends on: Date API (enabled), Locale (disabled)
✓	Date PHP4	8.x-2.7	Emulate PHP 5.2 date functions in PHP 4.x, PHP 5.0, and PHP 5.1. Required when using the Date API with PHP versions less than PHP 5.2. Depends on: Date API (enabled)
✓	Date Popup	8.x-2.7	Enables jquery popup calendars and time entry widgets for selecting dates and times. Depends on: Date API (enabled), Date Timezone (enabled)
✓	Date Repeat API	8.x-2.7	A Date Repeat API to calculate repeating dates and times from iCal rules. Depends on: Date API (enabled)
✓	Date Timezone	8.x-2.7	Needed when using Date API. Overrides site and user timezone handling to set timezone names instead of offsets. Depends on: Date API (enabled) Required by: Calendar (enabled), Date (enabled), Date Popup (enabled), Calendar iCal (enabled), Calendar Multiday (disabled), Date Tools (enabled), Calendar Popup (enabled)
✓	Date Tools	8.x-2.7	Tools to import and auto-create dates and calendars. Depends on: Content (enabled), Date (enabled), Date API (enabled), Date Timezone (enabled)

Figure 1. Calendar and date modules enabled

add new single-line textfield

Field settings

Category: *
Personal information
The category the new field should be part of. Categories are used to group fields logically. An example category is "Personal information".

Title: *
Company name
The title of the new field. The title will be shown to the user. An example title is "Favorite color".

Form name: *
profile_company_name
The name of the field. The form name is not shown to the user but used internally in the HTML code and URLs. Unless you know what you are doing, it is highly recommended that you prefix the form name with `profile_` to avoid name clashes with other fields. Spaces or any other special characters except dash (-) and underscore (_) are not allowed. An example name is "profile_favorite_color" or perhaps just "profile_color".

Explanation:
[Rich text editor interface showing a toolbar and a text area with the placeholder text "Enter the name of your company or organisation"]

An optional explanation to go with the new field. The explanation will be shown to the user.

Figure 2. Adding a field to the user profile

Figure 3. Modified sign-on screen

preferably contribute this to the community for others to use. While Drupal does provide excellent API support, the use of disparate *hard wired* hooks needs to be carefully considered.

In this tutorial I will approach a real life scenario I recently encountered developing a Drupal site.

The problem

Build me a booking system with a calendar was the request, and at first glance it seemed to be an relatively

Figure 4. Adding a new content type

Figure 5. Disable workflow, comments and images

straightforward challenge for Drupal, first create a custom content type called events and add any custom fields via CCK. Then report on the bookings using views and calender, and finally add the relevant permissions to prevent other subscribers seeing each others content, while at the same time showing the slot as being unavailable in the calender. As the time slots were a fixed duration and at certain fixed non-linear times, the only additional programming logic that would be required is *If slot empty => book, else => warn user and die*. Unfortunately, it was not as simple as that. First of all the time slots had to be of 45 minute duration and the date module only supports time increments of 1, 5, 15 and 30 minutes so and modifications to the date module would have major implications on the rest of the system (Don't hack core). Secondly, the gaps between the the slots were not linear (9:45, 10:30, 11:15, 12:00, 1: 15, 2:00, 2:45, 3:30) and needed to be easily changed in the future. Having looked at a number of booking/event/ timeslot modules on Drupal but I decided the best and most elegant solution was to create a custom field for the time slots and check that the content was unique using the `unique_field` module. This fulfilled the specification,

Figure 6. Edit the booking form

Edit
Manage tabs
Display fields

Booking form

Add fields and groups to the content type, and arrange them on content display and input forms.
You can add a field to a group by dragging it below and to the right of the group.

Label	Name	Type	Operations
+ Your reference	Node module form		
+ Menu settings	Menu module form		
+ Special instructions	Node module form		
+ Authoring information	Node module form		
+ Revision information	Node module form		
+ Publishing options	Node module form		
+ Printer, e-mail and PDF versions	Print module form		
+ Comment settings	Comment module form		
+ Path settings	Path module form		

Add

+ @ New field

Booking Date

Label

Valid from date

Field name (a-z, 0-9, _)

Date/time

Type of data to store:

Field with Date Pop-up calendar

Form element to edit the data:

+ @ New group

Label

group_

Group name (a-z, 0-9, _)

Save

Figure 7. *Adding a field to the booking form*

it keeps the interface straightforward, it validates at source the user input, prevents duplication, and is easy to maintain. There was only one fly in the ointment, the default error message displayed by the module allows the user to override the unique field values and as it is critical that only unique values are used, I had to amend the module by commenting out one line. There is an outstanding feature request for this functionality, so when I get some time I really should address this and submit a more appropriate modification to the source tree

The solution

Step 1 – Ensure appropriate modules are installed and active

Install the additional modules as detailed in Table 2. this is achieved by copying / SFTP'ing the tarball onto the server and extracting in the `/usr/local/www/drupal6/sites/all/modules` directory. JQuery comes in two parts, the Drupal

Booking form settings

This settings apply only to the Booking Date field as it appears in the Booking form content type.

Default value:

Now

A default value to use for this field. If you select "Relative", add details below.

Default value for To date:

Same as From date

A default value to use for this field. If you select "Relative", add details below.

[Customize Default Value](#)

Input format:

{27 Feb 2011}

Set the order and format for the date parts in the input form. The format will be adapted to remove values not in the granularity for this field.

Years back and forward:

[0 +3]

Number of years to go back and forward in the year selection list, default is -3 +3.

Time increment:


[1]

Increment the minute and second fields by this amount.

[Customize Date Parts](#)

Help text:

Choose the date of your booking



Instructions to present to the user below this field on the editing form.

Allowed HTML tags: <a>
 <code> <i> <pre> <sup> <sub> <div> <p>

CKEditor: the ID for including or excluding this element is admincontentnode-type-booking-field-date-edit-description.

Figure 8. *Choosing the date format*

Global settings

These settings apply to the *Booking Date* field in every content type in which it appears.

☒ Required

Number of values:

1

Maximum number of values users can enter for this field.

Unlimited will provide an "Add more" button so the users can add as many values as they like.

Warning! Changing this setting after data has been created could result in the loss of data!

To Date:

☒ Never

☐ Optional

☐ Required

Display a matching second date field as a "to date". If marked "Optional" field will be presented but not required; if marked "Required" the "to date" will be required if the "from date" is required or filled in.

Changing the "To date" setting after data has been created could result in the loss of data!

Granularity:

Set the date elements to be stored (at least a year is required).

Default Display:

Medium

Select a default format type to be used for the date display. Visit the [Date and time date format page](#) to add and edit format types.

Time zone handling:

No time zone conversion

Select the time zone handling method to be used for this date field.

Changing the Time zone handling setting after data has been created could result in the loss of data!

Figure 9. *Change the granularity for years – do not need time*

module (tgz) and the supporting JQuery code (.zip) which should be extracted into a directory called jquery.ui under the `jquery_ui` module directory. Enable all modules and the date modules as per Figure 1. If the trigger or profile modules are not enabled, enable them.

Step 2 – Create a custom content types

It would be good when our new users register that we have further details so we can contact them about their booking. This is achieved by adding the appropriate fields in *Home>Administer>User management>Profiles*. See Figure 2 and 3.

We now need to create a custom Event content type, which for this example will have a popup calendar field for the date, a slot time and a special instructions text area. Replace title with something more appropriate, like your reference, change the body title to Special Instructions and disable and unwanted functionality, e.g. comments. See Figure 4 – 5.

[illegible]

Figure 10. *Adding the time slot*



Figure 11. Time slot values

We now need to add a date field and a custom Slot field. Navigate to *Home>Administer>Content management>Booking form* and add a field for the date and a select list for the slot. Save your changes, and under the section Unique field settings ensure ensure that `booking_date` and `time_slot` are checked as a pair for for unique values. Navigate to *Home>Create content* and add some entries. Ensure no double bookings take place. If you wish to remove the option for users to override unique events, comment out the following code from `unique_field.module` thus:

```
// $msg .= '<p>'. t('Click !here to bypass this check and
resubmit.', array('!here' => „<a href=\"#\" onclick=\\
\"$('form#node-form input#edit-unique-field-override').val(1);
$('form#node-form').submit();return false;\\\">". t('here')
.\"</a>\")) . '<p>';
```

See Figure 6-15.

Step 3 – Create Views

Now we need to create a calender view and some reports. Navigate to *Home>Administer>Site building>Views* and create a new view called Subscriber bookings. Modify the following fields accordingly and create a page view with the url `subscriber_bookings`: see Table 1.

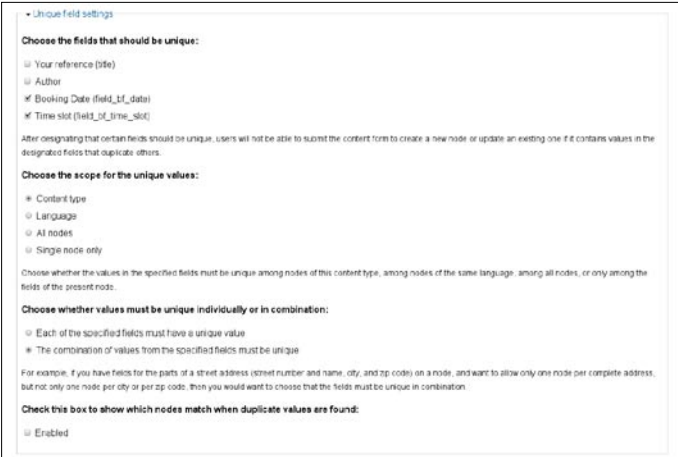


Figure 12. Preventing duplicate entries

Table 1. Subscriber bookings view settings

Basic settings	Style:Table Use pager: Mini Access: authenticated user
Fields	Profile: Company Details: Company name Company name Content: Booking Default Content: Time slot Default Node: Body Special Instructions
Filters	Node: Type= Booking form User: Current Yes
Page Settings	Path: subscriber_bookings

Table 2. Subscriber calender view settings

Basic settings	Access: authenticated user
Arguments	Date: Date (node) Content: Booking Date (field_bf_date)
Fields	Content: Booking Default Content: Time slot Default
Filters	Node: Type = Booking form User: Current Yes
Page settings	Path: subscriber_calender

Save your changes, clone the view and rename the path `subscriber_bookings_admin`, remove the user current field, change the access permissions as appropriate and save as `subscriber_bookings_admin`. This will allow administrators to see all bookings on the site, and normal subscribers to only view their content. As the Administrator profile does not contain the Company data, that filed is blank. The new profile for TEST was required to prompt for this information on registration. See Figure 16-20.

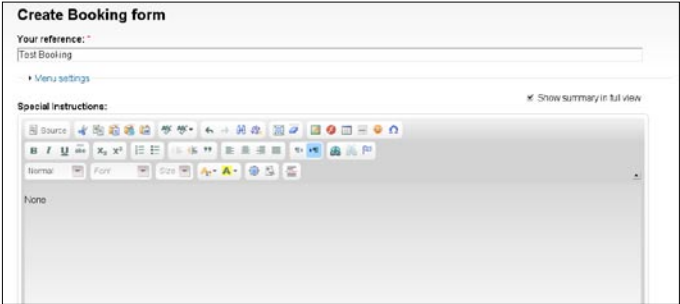


Figure 13. Adding a test booking



Figure 14. Picking the date

Table 3. Menu items and paths

Menu Item	Path
Add booking	node/add/booking
Calender of bookings	subscriber_calender
List all bookings	subscriber_bookings

Table 4. Email message for Admin and Subscriber

User	Body
Admin	A new booking has been made by %username.
Subscriber	Thank you for your booking. We will review and contact you if we experience any problems with your request.

Return to *Home>Administer>Site building* and clone the calender to `cal_bookings`. Amend the view as follows: see Table 2.

Repeat and create a new calender `cal_bookings_admin`, and change the path to `subscriber_calender_admin`. Remove `User:Current` Yes and update the permissions as appropriate. Save. See Screenshots 21-22.

Step 4 – Modify permissions and build menus

Navigate to *Home>Administer>User management* and create the appropriate role for the new user or alternatively

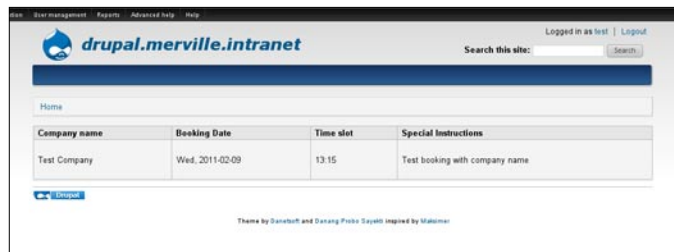


Figure 18. Subscriber view

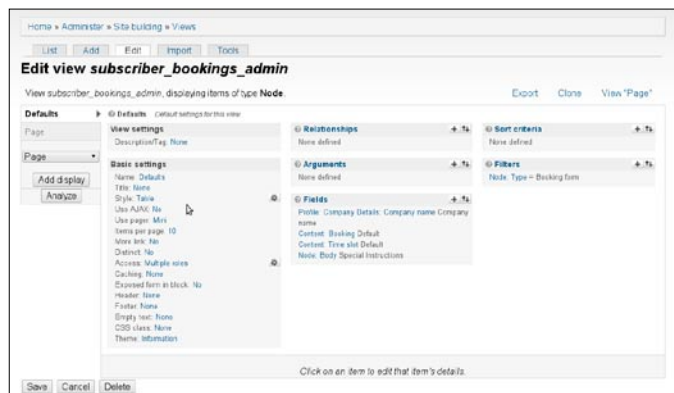


Figure 19. Subscriber_bookings_admin



Figure 20. Admin view before adding test data

just use the authenticated user role, but ensure that they do not have excess permissions. If you do change the role, you will need to update the access permissions in the 2 subscriber views.

Create a new menu called Subscriber menu and add links to Add Booking, View Calender and list all bookings as below: see Table 3.

Add the Subscriber menu to a block in your theme – in the default theme I have used (Danland) I have used the Superfish menu at the top. Configure the permissions as appropriate and save.



Figure 21. Admin view after adding test data

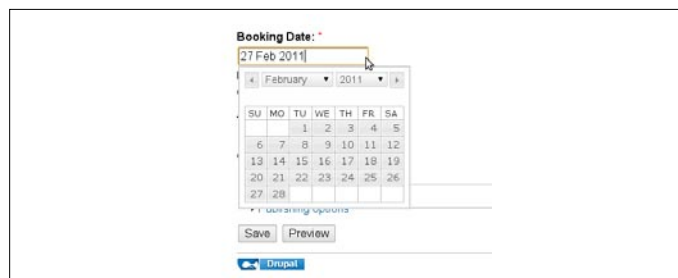


Figure 15. Javascript date popup

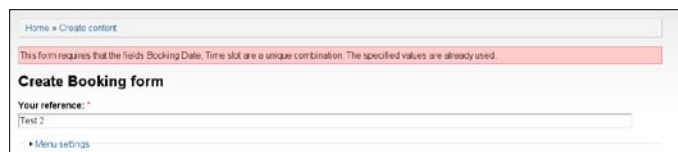


Figure 16. Error message on booking conflict

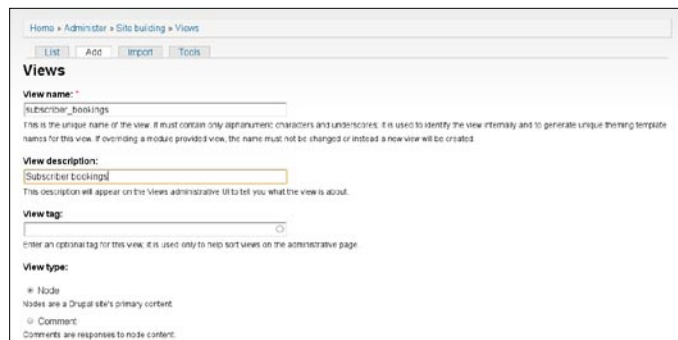


Figure 17. Creating a new view

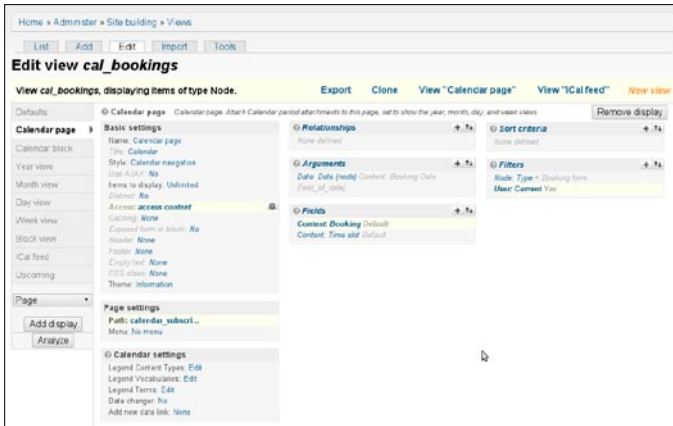


Figure 22. Cal_bookings view



Figure 23. Calander with slots booked

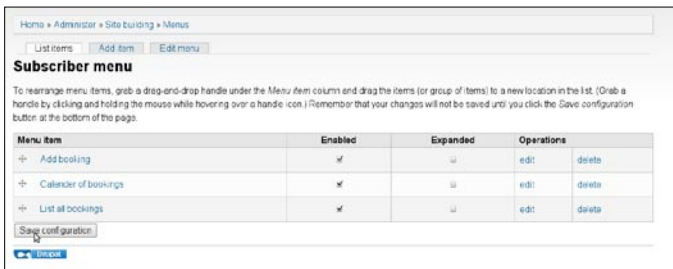


Figure 24. Bulding the subscriber menu

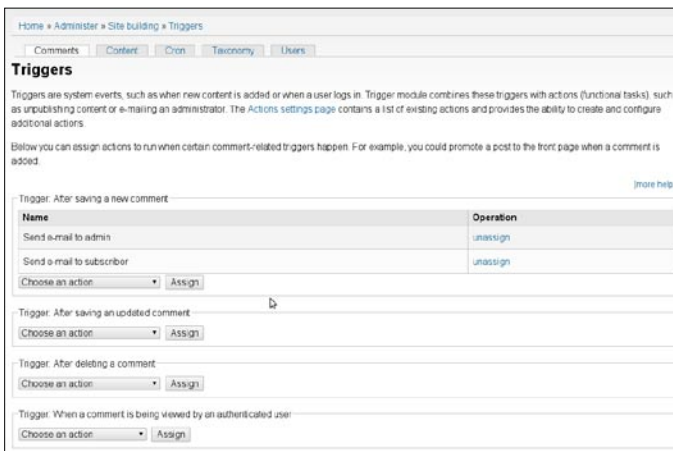


Figure 25. Triggers

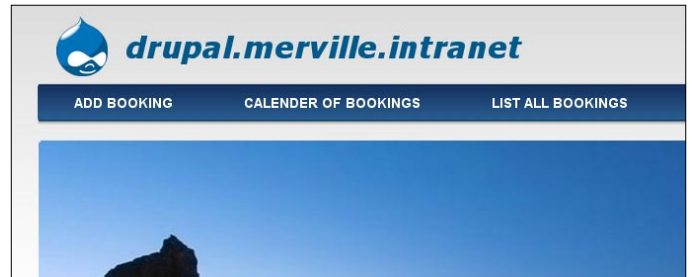


Figure 26. Superfish menu

On the 'Net

- <http://drupal.org/node/644854> – Request for unique error message in unique_fields
- <http://drupal.org/> – Drupal website

Additional Modules used

```
calendar-6.x-2.4.tar.gz
date-6.x-2.7.tar.gz
jquery_ui-6.x-1.4.tar.gz
jquery.ui-1.6.zip (Install in directory under jquery_
                    ui as jquery.ui)
unique_field-6.x-1.4.tar.gz
```

Step 5 – Notification email

We should notify our site manager of any new bookings by email, and our subscribers. Navigate to *Home>Administer>Site configuration* and add 2 new actions, Send e-mail to Admin and Send e-mail to subscriber. Id the destination field for subscriber use `%author:` see Table 4.

Navigate to *Home>Administer>Site building>Triggers >Comments* and add the admin and subscriber email to Trigger: After saving a new comment. Additional emails can be added as appropriate.

To do

Prevent user posting booking from before today (Filter view and warn user of bad input). Clean up Booking form and remove workflow fields etc.

ROB SOMERVILLE

*Rob Somerville has been passionately involved with technology both as an amateur and professional since childhood. A passionate convert to *BSD, he stubbornly refuses to shave off his beard under any circumstances. Fortunately, his wife understands him (she was working as a System/36 operator when they first met). The technological passions of their daughter and numerous pets are still to be revealed.*

BSDCAN 2011

THE BSD EVENT OF 2011
<http://www.bsdcan.org/>

Ottawa, Canada



BSDCan 2011 – The event to be at this year

BSDCAN 2011

There's only one major BSD Event in
North America in 2011: BSDCan

WHERE

Ottawa, Canada

WHEN

Early May 2011, with two days of tutorials
before the conference. Exact dates to follow.

WHO

Contributors, developers, and users

VENUE

University of Ottawa
<http://www.uottawa.ca/>

AT FEES YOU CAN AFFORD

We plan to keep costs to a minimum. As such, the conference will be held at University of Ottawa and accommodation is available within the University residences. Hotels are also within close walking distance of the conference venue.

WHAT DOES IT COST?

Type	CAD
Regular	\$195
Corporate	587
Additional Corporate	\$120
Student	57
Tutorial	57

Comfortable accommodation is available on campus at very reasonable rates. See our website for details.

Take the BSDA Certification exam.
For details see
<http://bsdcertification.org/>

SCHEDULE OVERVIEW

Wednesday

- 4:00 pm Sign-in desk opens at a local pub. Get your registration pack and have a drink.
- 8:00 pm Sign-in desk closes.

Thursday

- 9:30 am Opening words
- 11:00 am First set of talks
- 12:00 pm lunch
- 1:00 pm Second set of talks
- 2:00 pm break
- 2:30 pm Third set of talks
- 3:30 pm break
- 4:00 pm Fourth set of talks
- 5:00 pm Key Signing Party

Friday

- 10:00 am First set of talks
- 11:00 am break
- 11:30 am Second set of talks
- 12:30 pm lunch
- 1:30 pm Third set of talks
- 2:30 pm break
- 3:00 pm Fourth Set of Talks
- 4:00 pm Fifth Set of Talks
- 5:00 pm Closing words

Sat

- 8:30 am Breakfast
- 9:30-4:00 Tourist fun

TALKS FROM 2010

Please see the website for complete details.

- ClangBSD - Replacing gcc with clang
- Consideration for the BSD Professional Exam
- Security Implications of the Internet Protocol version 6 (IPv6)
- Puffy At Work -- Getting Code Right And Secure, The OpenBSD Way
- Everything you need to know about cryptography in 1 hour
- Networking from the Bottom Up: IPv6
- Porting dummynet to Linux and Windows
- Journaled Soft-Updates
- Porting hwpmc to non x86 platforms
- Maintaining a Customized FreeBSD Distribution
- Debuggers - Architecture and Implementation
- pfSense 2.0
- Networking from the Bottom Up: Ipv6
- The New VWorld
- A new packet scheduling architecture for FreeBSD

SOCIAL ACTIVITIES

It's not all work. Social activities play a major role in project development.

Wednesday

4:00 pm Drinks + registration at a local pub

Thursday

4:30 pm BOFs

7:00 pm Gathering at local eateries for dinner

Friday

4:30 pm Key signing party

7:00 pm Gathering at local pubs for drinks

Saturday

8:30 am Breakfast

afterwards: various tourist-type things

To stay informed, please join our announcement mailing list. Details at <http://www.bsdcn.org/>

2010 PLATINUM SPONSOR



OUR 2010 GOLD SPONSORS



The FreeBSD Foundation Logo is the trademark of the FreeBSD Foundation

Using FreeBSD

to authenticate users with OpenLDAP and FreeRADIUS

We introduce a WIFI authentication environment using 802.1X with a RADIUS server (FreeRADIUS), a central database (like OpenLDAP) to store user and password, and using MSCHAPv2 protocol to avoid third party supplicants.

What you will learn...

- Install and configure FreeRADIUS
- Configure FreeRADIUS with OpenLDAP authentication
- Configure the Access Point to work with FreeRADIUS

What you should know...

- Basic use of ports system
- OpenLDAP operation
- Configure a generic Access Point

Objective

Create an environment to authenticate users against a database or OpenLDAP, using 802.1x protocol, with FreeRADIUS as RADIUS server. The main idea isn't use third-party supplicant, the explanation about why, are related to make no or minimal modifications on client operational systems.

For this reason, we will use MSCHAPv2 to authenticate users, because Linux, FreeBSD, MacOSX and Windows are compatible with this challenge-response protocol.

To make things easy, we need to explain a little thing, OpenLDAP needs *sambaNTPassword* and *samba.scheme* working on OpenLDAP, and more, only ClearText on *userPassword* attribute or hash NT on *sambaNTPassword* will work on this environment. The NT/LM password work for simple reasons, but, using ClearText, FreeRADIUS can create the NT hash and start the challenge-response authentication. On simple words, any other hash will not work (SHA1, MD5, Crypt) or any other non descriptable hash.

Environment

The environment will need some items, systems and equipments to work properly:

- One server running FreeBSD 8.2: IP: 200.129.192.94
- FreeRADIUS 2.1.10 (installed using ports)

- OpenLDAP 2.4.23 (to authenticate)
- AP 3COM 7760 (you can use any other with support):
IP: 200.129.202.132
- Domain: *ufms.br*

OpenLDAP needs to be working properly with *samba.scheme* support, the 3COM access point are used, because support radius authentication using WPA2 Enterprise with AES encoding, in this case, you can use what you want, the only restriction here is the support to RADIUS authentication. I show this IPs to make easier to explain and the domain are used to generate certificates to server.

How it works, the users will connect on AP and will use WPA2 Enterprise with the radius server configurated to authenticate using 802.1x and on second step, using PEAP/MSCHAPv2 to authenticate with FreeRADIUS. Of course, a better option against MSCHAPv2 will be EAPTTLS/PAP with third party supplicant, but for users are more simple use the autoconfiguration of your operational system.

Our users will be on OpenLDAP that will receive FreeRADIUS connections to request user informations using a secure channel with TLS.

Installation Procedure

We will use the more recent *port* of FreeRADIUS, so let's search this package:


```
# cd /usr/ports
# make search name=freeradius display=name,path
Port:    freeradius-2.1.10_2
Path:    /usr/ports/net/freeradius2
```

Change to this directory:

```
# cd /usr/ports/net/freeradius2
```

And on this step, we will configure OpenLDAP support for FreeRADIUS:

```
# make config
```

On configuration screen, mark the LDAP support for FreeRADIUS:

```
[X] LDAP                With LDAP database support
```

At this point, compile and install this package:

```
# make install clean
```

With our system clean, all the dependencies will be installed, like any other port. For a clean system this is the list of dependencies:

- perl
- python26
- libiconv
- m4
- openldap-client (2.4.23)

Right now, FreeRADIUS is installed.

FreeRADIUS Configuration

After the installation of FreeRADIUS, we will do some steps to avoid errors, so, let's configure a simple equipment user using the main file of users of FreeRADIUS:

```
# cd /usr/local/etc/raddb
```

Inside this directory are all files needed to configure FreeRADIUS, edit the file: see Listing 1.

This file contains information about who can authenticate using the radius server, at this file we append our AP (with IP 200.129.202.132), so, the AP will only do connections with FreeRADIUS because this configuration (don't forget to enable others APs, or all the subnet instead of IP). Remember, this clients are the equipments, not the people.

To test our configuration, now we will configure a *real user*, and we can test the connection without use the AP, instead using the command `radtest`, let's create the user:

```
# ee /usr/local/etc/raddb/users
...
„John Doe”      Cleartext-Password := „hello”
                  Reply-Message = „Hello, %{User-Name}”
```

This example is suggested by FreeRADIUS, and in this case, the user is *John Doe*, and the password is *hello*, let's test the connection:

```
# radtest -t pap „John Doe” „hello” localhost 1812 testing123
```

You will see anything like: see Listing 2.

Now, execute on another terminal the FreeRADIUS in debug mode:

```
# radius -X
```

Listing 1. Configuring AP as client

```
# ee /usr/local/etc/raddb/clients.conf
client localhost {
    ipaddr = 127.0.0.1
    secret      = testing123
    require_message_authenticator = no
    nastype     = other      # localhost isn't
                             usually a NAS...
}

client 200.129.202.132 {
    secret      = password_set_on_radius_server
    shortname = ap-radius
}
```

Listing 2. Log radtest request ClearText

```
Sending Access-Request of id 116 to 127.0.0.1 port
1812
    User-Name = "John Doe"
    User-Password = "hello"
    NAS-IP-Address = 127.0.0.1
    NAS-Port = 1812
rad_recv: Access-Accept packet from host 127.0.0.1 port
1812, id=116, length=37
    Reply-Message = "Hello, John Doe"
```

This mode is used to debug FreeRADIUS, and now, when you test the authentication with *radtest* you can see on logs:

```
[pap] login attempt with password „hello”
[pap] Using clear text password „hello”
[pap] User authenticated successfully
```

It's shows that the user was successfully authenticated. If you want to test the connection using this user with AP, it will work too.

Now you will configure the OpenLDAP's connection. Edit the *ldap* module: see Listing 3.

Active the TLS only with your OpenLDAP support it, you can test without, but I recommend the encrypted connection. Using the radius in debug mode you will see:

```
[ldap] attempting LDAP reconnection
[ldap] (re)connect to ldap.ufms.br:389, authentication 0
[ldap] starting TLS
```

At this time, our FreeRADIUS is connected with success with OpenLDAP using TLS. To do a real authentication or OpenLDAP need users with some attributes, you can use *userPassword* with ClearText, or you can use a

Listing 3. Configure OpenLDAP settings

```
# ee /usr/local/etc/raddb/modules/ldap

ldap {
    server = "openldap.ufms.br"
    basedn = "dc=ufms,dc=br"
        identity = "cn=user_for_read,dc=ufms,dc=br"
        password = password_of_reader_on_ldap
    filter = "(uid=%{%Stripped-User-Name}:-%{User-
        Name}})"
    ldap_connections_number = 5
    timeout = 4
    timelimit = 3
    net_timeout = 1
    tls {
        start_tls = yes
        require_cert = "allow"
    }
    dictionary_mapping = ${confdir}/ldap.attrmap
    edir_account_policy_check = no
}
```

Listing 4. *OpenLDAP user with NT password*

```
# ee user.ldif
dn: uid=test,dc=ufms,dc=br
sn: do Test
cn: Test do Test

objectClass: person
objectClass: inetOrgPerson
objectClass: sambaSamAccount
userPassword: {SSHA}gWRX6IuyiGw+0xvPN3JhaGEcvuLJqm1B
sambaNTPassword: 1E39A9A92F2B08A0E69B4D5ADA7E5332
sambaSID: 1
```

Listing 5. *Log radtest request NT password*

```

Sending Access-Request of id 151 to 127.0.0.1 port
    1812
User-Name = "test"
NAS-IP-Address = 200.129.192.94
NAS-Port = 1812
MS-CHAP-Challenge = 0x2ff26066cb1a2416
MS-CHAP-Response = 0x00010000000000000000000000000000
                    00000000000000000000000006f252f352fd4
                    c0af86d8c3737866243af03519ca145886
                    6f
rad_recv: Access-Accept packet from host 127.0.0.1 port
    1812, id=151, length=84
MS-CHAP-MPPE-Keys = 0x000000000000000005610a3a37fccb
                    de5c7d37764aa0b979300000000000000
                    0
MS-MPPE-Encryption-Policy = 0x00000001
MS-MPPE-Encryption-Types = 0x00000006

```

Listing 6. *FreeRADIUS MSCHAPv2 success*

```
[peap] Got tunneled reply RADIUS code 2
      MS-MPPE-Encryption-Policy = 0x00000001
      MS-MPPE-Encryption-Types = 0x00000006
      MS-MPPE-Send-Key = 0x832ff5d837c847d30e40883b
                        94d6d02d
      MS-MPPE-Recv-Key = 0xb104726dfdd1dd050a2db359
                        fa016836
      EAP-Message = 0x03080004
      Message-Authenticator = 0x00000000000000000000
                        000000000000
      User-Name = "test"

[peap] Tunneled authentication was successful.

[peap] SUCCESS
```

little more secure (but considered ClearText too) NT/LM, using *sambaNTPassword*. For an example, you can use this *ldif* user: see Listing 4.

The password used is *senha1*, so you use a {SHA1} password for other systems, and NT password for FreeRADIUS authentication.

When you try to authenticate with this user:

```
# radtest -t mschap test senha1 localhost 1812 testing123
```

You can see on the user's side: see Listing 5. And this on the server side: see Listing 6.

Your user was successfully authenticated. The last thing is to make FreeRADIUS start automatically with FreeBSD boot, edit the *rc.conf* file:

```
# /usr/local/etc/rc.d/radiusd rcvar >> /etc/rc.conf
```

And modify the *radiusd_enable*:

```
# ee /etc/rc.conf
```

```
...
radiusd_enable="NO"
...
```

to:

```
radiusd_enable="YES"
```

Now your system was configured properly.

Access Point (AP) Configuration

To configure the AP we only need to point the FreeRADIUS Server IP, the port and the password we defined in *users* file of FreeRADIUS. We edit one of the VLANs with our configuration: Figure 1.



Figure 1. AP-Wifi System

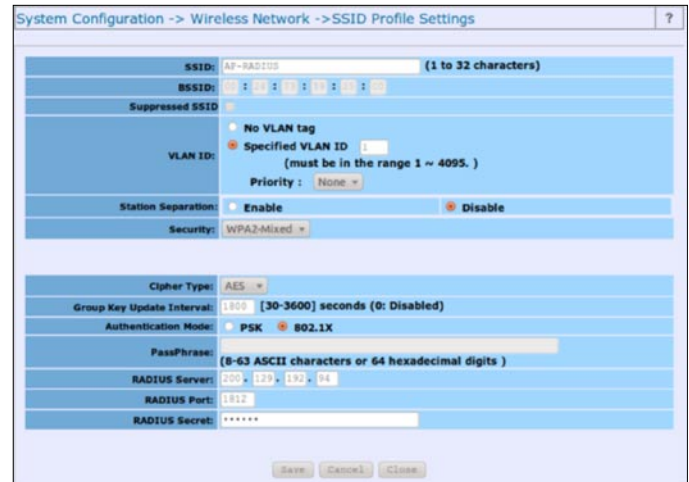


Figure 2. AP-Profile Settings

Choose *edit* to configure the first VID, we use AP-RADIUS on SSID, and WPA2-Mixed aka Enterprise WPA/WPA2 with AES Cipher: Figure 2.

The main informations you can do attention:

- SSID: AP-RADIUS
- Security: WPA2-Mixed
- Cipher Type: AES
- RADIUS Server: 200.129.192.94
- RADIUS Port: 1812 (default)
- RADIUS Secret: *password_set_on_radius_server*

This is the only thing you need to configure in your AP. We use IAPP for wifi migration, but this is not in the scope of this paper.

Client Configuration Example

Table 1. Table of Clients Compatibility

Vendor				
Microsoft	Windows XP SP3	Windows Vista	Windows 7	
Apple	MacOSX Snow Leopard	MacOSX Lion	iOS 4.2	iOS 4.3
Linux	Debian Squeeze	Ubuntu 10.04 LTS	CentOS 5.5	
BSD	FreeBSD 7.3	FreeBSD 8.1	PCBSD 8.2	
Google	Android 2.1	Android 2.2	Android 2.3	Android 3.0

Table of Clients Compatibility

This table was created using our configuration above as tested, the FreeRADIUS of course can support many others, but with this we can guarantee working properly.



Figure 3. MacOSx-wifi-choose

The systems use MSCHAPv2 with minor modifications are possible.

The green represents working systems, and the gray represent, untested systems but expected to work without problems.

Client Configuration Example: MacOSX Snow Leopard

The configuration made on MacOSX Snow Leopard is simpler than the configuration on iOS or Windows, select the AP-RADIUS WIFI network: Figure 3.

Insert user and password (the MacOSX will choose the best authentication mode for 802.1X): Figure 4.

And after that, accept the certificate, the MacOSX will warning you, because the certificate is auto signed, but this was expected. Click on *continue* button: Figure 5.

The MacOSX will insert the main certificate on your keys and you don't need to accept this anymore.

Conclusion

This paper was made thinking on how to create a simple VLAN for students of an University in Brazil, to use the Internet (like EDUROAM) only inside the institution without lose your connection (IAPP) and to use a better option to authenticate for using WIFI than share the WPA/

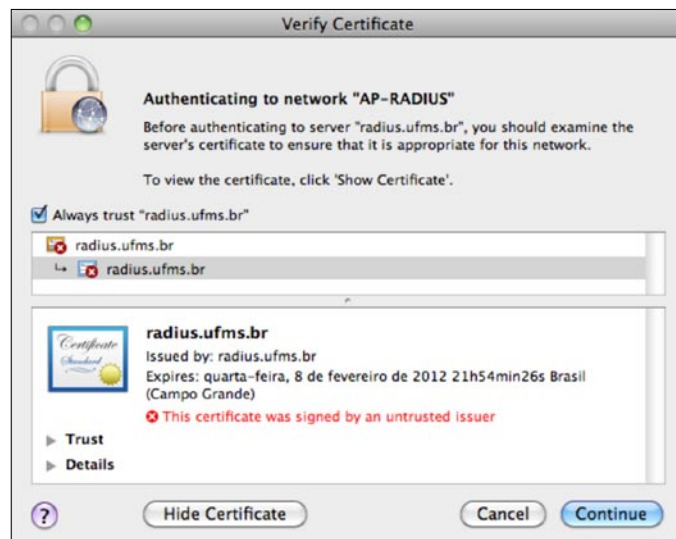


Figure 5. MacOSx-certificate

On the 'Net

- <http://freeradius.org/>
- <http://www.freebsd.org/>
- <http://www.bibliotecaunix.org/>

WPA2 with each student. We expect this work helps other institutions that need an option to authenticate users on a centralized directory or database.

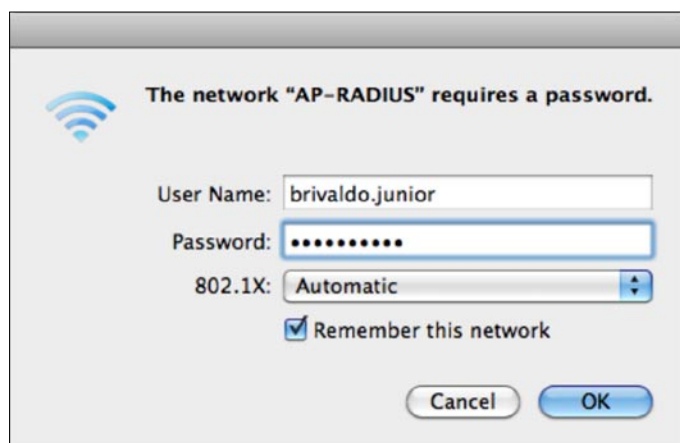


Figure 4. MacOSx-user-password

BRIVALDO JUNIOR

Brivaldo Junior holds a BS in Computer Science, currently is Master Degree student in Networks, and works as head of the Networks Division at the Federal University of Mato Grosso do Sul. Enjoys open technologies such as Linux and BSD and maintains a blog in Portuguese about Unix in general.
condector@gmail.com



Data **CENTER** FOR IT PROFESSIONALS MAGAZINE

Want to have all the issues of Data Center magazine?
Need to keep up with the latest IT news?
Think you've got what it takes to cooperate with our team?

Check out our website and subscribe to Data Center magazine's newsletter!

Visit: <http://datacentermag.com/newsletter/>

How To Setup OpenBSD On The Embedded Alix Card

In this article you will learn how to Install an operating system on an ALIX card. It's a an invaluable tool for a System Administrator. Following this guide will help protect your internal network from the hostile Internet!

What you will learn...

- How to install OpenBSD on embedded device, in this case on an ALIX card.

What you should know...

- How to install OpenBSD

What is ALIX ?

ALIX (<http://www.pcengines.ch/alix.htm>) is a small (6x6inch), low power motherboard. It's a perfect device for home or business firewall application.

Embedded on the ALIX is a Geode (i386 compatible) processor. So you can install a lot of different OS. But with OpenBSD you can maximize its full potential.

My card: <http://www.pcengines.ch/alix6e1.htm>.

What do you need

- A computer with an Internet connection
- An ALIX board
- A RS-232 serial cable between your computer and the ALIX board
- A RJ45 cable between your computer and the ALIX board

Your computer will be used to provide DHCP server and tftp server for the PXE boot of the ALIX card. For this paper, my computer is running an OpenBSD 4.8:

```
uname -a
OpenBSD laptop.my.domain 4.8 GENERIC#136 i386
```

Vocabulary

- PXE server: An OpenBSD 4.8 laptop with a DHCP and tftp server installed

- PXE client: The ALIX card.

A PXE server is composed of two things:

- A dhcp server: To give an IP configuration to the ALIX card during the boot process and the filename of the kernel that will be loaded via tftp
- A tftp server: To send to the ALIX card the kernel Installation

My configuration for this installation

See Figure 2.

Installation of DHCP server on your laptop

Add correct source for pkg:

```
export PKG_PATH=ftp://ftp.fr.openbsd.org/pub/OpenBSD/
4.8/packages/i386/
```

Installing the server:

```
pkg_add -iv isc-dhcp-server
```

Configuration of dhcp server

Create a configuration file like this, in `/etc/dhcpd.conf`

```
option domain-name-servers 192.168.1.254;
```



```
subnet 10.0.0.0 netmask 255.0.0.0 {
    option routers 10.0.0.254;

    range 10.0.0.10 10.0.0.20;

    filename „pxeboot“;
}
```

Start dhcp server

dhcpcd

Activation of TFTP server on your laptop

You don't need to install it, but just activate it.

Edit the file `/etc/inetd.conf` and uncomment this line:

```
tftp          dgram    udp      wait    root    /usr/
              libexec/tftpd    tftpd -s /tftpboot
```

Then, create the directory for tftp service

```
mkdir /tftpboot
```

And download the required executable for the PXE boot process in the proper folder:

```
cd /tftpboot
ftp ftp://ftp.fr.openbsd.org/pub/OpenBSD/4.8/i386/pxeboot
ftp ftp://ftp.fr.openbsd.org/pub/OpenBSD/4.8/i386/bsd.rd
```

Note

The ALIX's CPU is Geode, which means it is i386 based.

Restart inetd to enable tftp

```
kill -HUP `cat /var/run/inetd.pid`
```

Enable NAT on your laptop

You can configure NAT on your laptop to give an Internet access to your ALIX card during the installation to get the sets.

Enable routing

```
sysctl net.inet.ip.forwarding=1
```

Enable NAT on PF

Edit `/etc/pf.conf` and write (adapt to your device and networks)

```
pass out on r10 from 10.0.0.0/8 to any nat-to
192.168.1.108
```



Figure 1. The Hardware

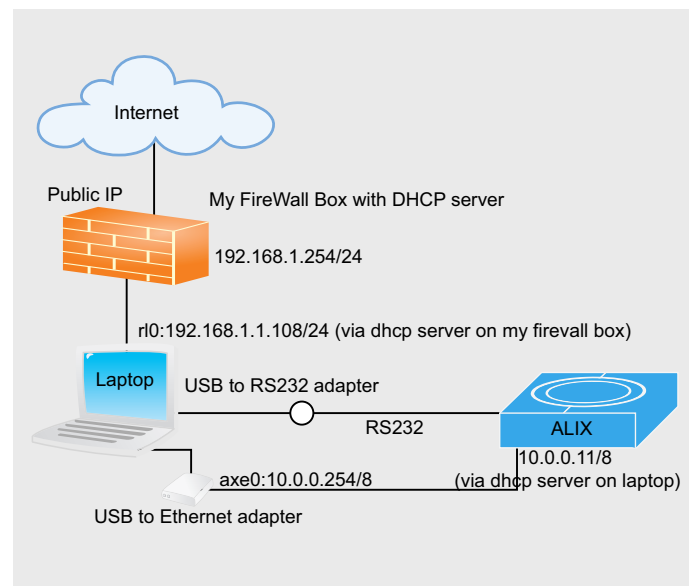


Figure 2. Setup configuration

Listing 1. ALIX booting OpenBSD in PXE

```
PC Engines ALIX.2 v0.99h
640 KB Base Memory
261120 KB Extended Memory

01F0 Master 044A CF 1GB

Phys C/H/S 1966/16/63 Log C/H/S 983/32/63

Intel UNDI, PXE-2.0 (build 082)
Copyright (C) 1997,1998,1999 Intel Corporation
VIA Rhine III Management Adapter v2.43 (2005/12/15)

CLIENT MAC ADDR: 00 0D B9 1C 9A 60
CLIENT IP: 10.0.0.10 MASK: 255.0.0.0 DHCP IP:
           10.0.0.254
GATEWAY IP: 10.0.0.254
probing: pc0 com0 com1 pci pxe![2.1] mem[640K 255M
          a20=on]
disk: hd0+*
net: mac 00:0d:b9:1c:9a:60, ip 10.0.0.10, server
      10.0.0.254
>> OpenBSD/i386 PXEBOOT 3.15
boot>
```

Enable PF

```
pfctl -ef /etc/pf.conf
```

Preparation of the ALIX card

We need to view what is happening on this card, and we can do it via RS232 cable.

In my case I use an USBtoRS232 adapter because my laptop like most modern laptops, does not have a built-in RS-232 connector.

You will need a software to connect to your RS-232 serial port. We can use minicom.

Installation of minicom on your laptop

```
pkg_add minicom
```

Configuration of minicom

```
minicom -s
```

- Go to *Serial port setup*
- Press **A** and write your device.
- For me with the USBtoRS232 it's `/dev/ttyU0`.

- Press **E** for speed and press **G** for 38400.
- It could be 38400 8N1 (press **Q** if not)
- Press **ENTER** twice and select *Exit*.

Now you can power up your ALIX board and see the boot process on your laptop screen !

Enable the PXE boot on the card

While the memcheck is running, press **S** key to print a minimal BIOS setup.

Then, press **e** key to enable the PXE boot and **q** key to save and quit the BIOS.

Remember to disable the pxe boot when your system will be installed to avoid reinstalling the software when you reboot your system.

ALIX booting in PXE mode

See Listing 1. Write this just after "boot>"

- boot> stty com0 38400
- boot> set tty com0
- boot> bsd.rd

The rest of the installation is standard except of when the installer asks *Change the default console to com0?*, say Yes:

- Change the default console to com0? [no] yes
- Available speeds are: 9600 19200 38400 57600 115200.
- Which one should com0 use? (or *done*) [38400]

This way for the next boot, your system will redirect the output to tty and not default screen.

GUILLAUME DUALÉ

Guillaume Dualé (g.duale@otasc.org) is a System and Network Administrator specialised in free-software.

He reside in south of France, he love BSD and GNU/Linux systems.

Looking for help, tip or advice?
Want to share your knowledge with others?

BSD MAGAZINE

BSD

Give us your opinion about the magazine's content
and help us create the most useful source for you!

Setting up Git and Mercurial Servers

GitHub provides an excellent web-based interface to Git with extensive project management tools. Bitbucket provides an equally excellent web-based interface for Mercurial.

What you will learn...

- How to configure permissions on Git and Mercurial servers
- How to manage users and groups for DVCS platforms
- Conceptual differences in managing DVCS from CVS and Subversion

What you should know...

- How to install applications
 - How to manage users, groups, and file permissions
 - How to use Git and Mercurial
-

However, project requirements, management concerns, or security needs may prevent the use of public storage tools for distributed version control. Under these circumstances, both Git and Mercurial are easy to set up and use on a BSD-based server. The niceties of the web interfaces are lost, but the full power of both *distributed version control system* (DVCS) platforms are available at the command line.

This article outlines the basic directory and permissions structure necessary to maintain a Git or Mercurial server on a BSD platform and accessible over SSH. However, this article assumes you are already familiar with how DVCS platforms operate and with server and SSH operations.

In addition, this article assumes you are familiar with installing applications through the ports and package systems, as appropriate, for your operating system. In general, these tips are equally valid on other Unix-like platforms, as well.

Incidentally, there is no reason not to manage both Git and Mercurial servers on a single server. The two DVCS platforms operate independently of each other and do not interfere with each other. This is valuable if local conventions cannot be mandated and cooperation with external entities mandates working with both Git and Mercurial. Because Git and Mercurial repositories ultimately form a mesh or star network of patches

and forks, working with an external repository can be aided by maintaining a local server which centralizes synchronization.

Installation

Unlike some systems, neither Git nor Mercurial require separate servers in the usual sense. Both can operate over SSH and HTTP. Git can also transport version control information over a native protocol, but this protocol's server is bundled directly into the Git client. However, both require their respective client to be installed on the server to operate it. Because of this, installation on a BSD-based server is as simple as installing the clients. Both Git and Mercurial can be installed using your BSD's native application packaging system or can be configured and installed directly from the package distributions provided by each development group.

Of note, Git is mostly C language and consists of many different programs each of which provides small parts of program's subcommands. Some are implemented in Perl and as shell scripts. In contrast, Mercurial is pure Python and requires a complete Python installation as a result. Both are relatively easy to install when using the native packaging system.

A Repository Home

One of the key aspects of both Git and Mercurial is how they store their repositories. If you are familiar with CVS

or Subversion, these turn version control on its ear. For CVS and Subversion, the working copy after a checkout is an image of the repository at a certain point in time. The history is stored in a central location. DVCS systems change this by packaging the history with each copy of the repository.

With CVS and Subversion, the server copy is special and cannot be treated as a working copy. A Git or Mercurial *server* is a copy of the repository just like any other, though the local checkout may not be present. Because of this, a Git or Mercurial central repository requires minimal planning and foresight. Indeed, the idea of a central repository in Git and Mercurial is more of a social convention than something technically enforced.

The first question to answer is where will storage of these repositories be kept. It is not unreasonable to store them with user accounts under `/home`, using `/home/git` and `/home/hg` for each. Given the nature of source code repositories, storing them under `/var` or `/var/db` is also reasonable. In this case, I have used `/var` for both repositories leading to the directories `/var/git` and `/var/hg`.

In each case, I created symbolic links from `/git` to `/var/git` and `/hg` to `/var/hg`. This shortening will be useful in creating remote paths. When tunnelling Git over SSH, paths are mapped one-to-one and shorter paths are desirable. With symbolic links in place, the path becomes `user@host:/git/repo`. Repositories on other locations can be accessed in the usual way, with one in howardjp's home directory being addressed as `user@host:/home/howardjp/repo`.

Mercurial offers the same advantage, but with a slightly different nomenclature. When using SSH, Mercurial requires a protocol specification that Git does not, so SSH-tunnelled Mercurial connections resemble `ssh://user@host//hg/repo`.

Managing Repository Permissions

Repositories themselves are managed in the tradition BSD way. In my example, I have created two user accounts to manage these storage areas. From `/etc/passwd`:

```
git:*:902:99:Git Repository Owner:/var/git:/usr/sbin/
nologin
hg:*:903:99:Mercurial Repository Owner:/var/hg:/usr/sbin/
nologin
```

Like all properly managed role accounts, these accounts are disabled through the use of an asterisk in the password field. Additionally, both have their shells set to `nologin`, which automatically disconnects a user when

launches the shell. The only purpose of these accounts is to own the parent directory for repositories and they could be merged into one account, if that is the local preference.

The group number listed, 99, is a group called `src`, which is otherwise unremarkable. Any group name and number will do. Users can be added to the `src` group to give them access to both Mercurial and Git repositories. Further restrictions of access are possible with the usual BSD group mechanisms. If ACLs are available due to special filesystem capabilities, they will be honored, as well.

But if a repository is meant to be shared among multiple users, it should have its permissions set appropriate to ensure all necessary users share read and write access correctly. The logic way to manage this is by setting the group on a repository to a project's group and making the repository readable and writable by the group. This must be done recursively on all files in the repository directory.

Users familiar with administering CVS central repositories can lock down individual components within the CVS tree and mark off sections of the tree for editing by some users through BSD's permissions structure. With both Git and Mercurial (and, incidentally, Subversion), this type of restriction is not possible. Git and Mercurial use an internal database format for storing changes leading to an all or nothing permissions situation. Environments which require multiple sets of editing permissions on repositories are best off dividing projects into multiple repositories.

Conclusions

These basic steps will help ensure a smoothly running and easier to maintain Git or Mercurial server. However, these tips cannot address every possible issue or local configuration requirement you may encounter in building a Git or Mercurial server. But these tips will provide the foundation for a sound server installation for DVCS platforms. Fortunately, unlike other popular version control systems, Git and Mercurial will continue functioning when the server is unavailable allowing the opportunity to fix mistakes.

JAMES P. HOWARD, II

The author is a senior analyst in Washington, DC, in the United States where he focuses on statistical and mathematical systems. He can be reached at jh@jameshoward.us or via Twitter [@howardjp](https://twitter.com/howardjp).

The Wonders Of Blender

Blender is a powerful software, but can also be daunting, especially for BSD users, as the award-winning software isn't yet officially favored on BSD. Fear not! Let's explore this wonderful tool, starting with the user interface.

What you will learn...

The article focuses on introducing Blender to BSD users. The readers are expected to gain knowledge about 3D design, the Blender software in general and game/movie/basic shapes in particular. Further more, additional expertise shall be provided on meshes, vertices, lamps, lights, nodes, raytracing, viewports, etc.

What you should know...

Basic knowledge of design will be required, such as acquaintance with geometrical 2D/3D shapes. The article assumes no prior expertise with any other 3D modelling software, yet, dexterity with the mouse and/or other similar device shall come in handy. In the game engine section, familiarity with Game Physics is beneficial though not vital.

The User Interface

Blender is a free, powerful and open source 3D graphics program. Released under the GNU GPL, it is available for multiple operating systems including Windows, Mac OS X and GNU/Linux. Wondering about BSD? Well yes, the Solaris builds run perfectly well, and it is also available via ports. Blender has held the distinction of having an easier learning curve for experts and newbies alike, as compared to other confusing and complex proprietary 3D softwares.

Now, getting straight to business! In this first leg of the tutorial, we shall cover the essential facts about the Blender interface.

The latest release of Blender is version 2.5. However, for learning purposes, I would recommend you to opt for version 2.49b as it has the most extensive documentation to its credit and is considered to be the most stable build thus far (for BSD, that is).

User Interface and Layout

Once you've installed Blender, it is time to run it! Blender is meant to run in fullscreen by default, though a windowed mode is also present.

When working with 3D models, you will need to switch between one viewport (also called window) to another. By default, the Blender interface consists of the following:

1. **3D Viewport:** It refers to the large mid-section of the interface. This is where you will view and work with 3D objects.

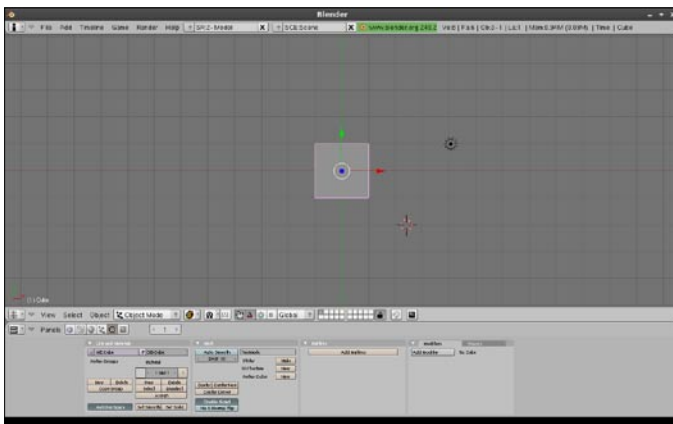


Figure 1. The blender interface

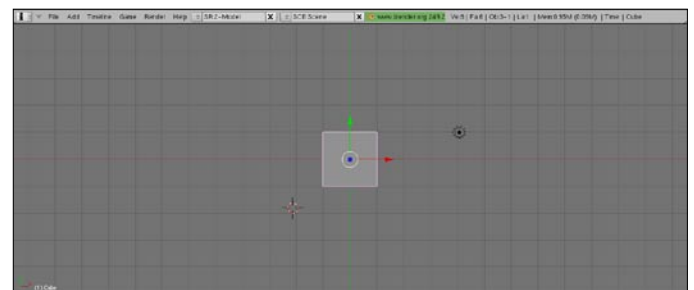


Figure 2. The 3D viewport (default view)

2. **Buttons Window:** Buttons allow you to edit, manipulate and alter the objects visible in the 3D Viewport.
3. **User Preferences Window:** Its header is shown at the top-most section of the interface (Figure 1).

In Blender, almost all the functions have a direct keyboard shortcut to facilitate working. Plus, the various parts of the interface are all drawn in OpenGL and thus can be handled much the same way as one would deal with 3D elements. Therefore, you can zoom in and out of GUI buttons like you would with contents of the 3D Viewport.

Blender has two main work modes: *Object Mode* and *Edit Mode*. Object Mode is used to edit entire objects (such as a complete model of a rectangle) while Edit Mode is used to work with individual components of objects (like individual vertices of a rectangle). <Tab> key is used to toggle between the two modes.

The 3D Viewport

Blender's 3D Viewport is where all the action happens, so let's first cover this section in detail (see Figure 2).

Movement in the 3D Viewport is controlled by the mouse and the Num Pad on the keyboard. The basic numeric keys you should bear in mind are 7, 1 and 3 for top, front and right views respectively. Placing the cursor anywhere in the viewport and typing these numbers takes you to the appropriate view. By default, 0 refers to the centered camera view (see Table 1 and Figure 3).

The left-click on mouse is used for selecting and dragging in object mode while the right-click is used in edit mode (more on the modes in next part of the tutorial). The scroll wheel on the mouse is used to zoom in and out. Be aware of the fact that the numeric keys refer to only those on the Num Pad, not the ones above the alphabetical keys!

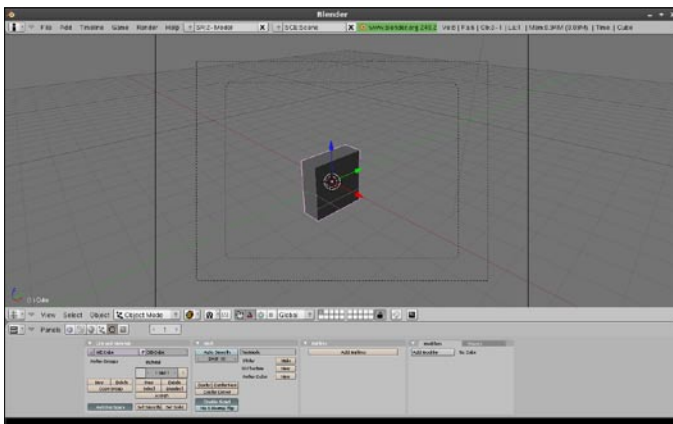


Figure 3. Moving around the 3d viewport

The best way to get accustomed to the keyboard shortcuts is to experiment and use them as frequently as possible.

Buttons Window

Traditionally, the Buttons window is placed at the lower portion of the screen. It consists of several buttons and each button has its own subset of functions. The buttons are as follows:

1. **Logic Button:** This is mainly used for game engines and activated by using F4.
2. **Script Button:** This connects the various events to scripts for complex projects and models.
3. **Shading Button:** It consists of sub-functions to control light, opacity, color, texture and other related settings. It is activated by using F5.
4. **Object Button:** As the name suggests, it activates commands for working with objects and is activated via F7.
5. **Edit Button:** It is used to edit object components in edit mode and is activated by using F9.
6. **Scene Button:** It is meant for rendering (still images) and animating (movies) and is activated via F10 (see Figure 4).

Once you click on a button, you will notice a set of numerous functions associated with it. For example, the given figure shows the Shading Panel (Figure 5).

User Preferences Window

This window is hidden by default, and contains some least used features. To make it visible, click and drag the header or Menu bar at the topmost area of the program downwards (Figure 6).

Table 1. Keyboard shortcuts in blender

Key Combo	Action
Numpad 5	Toggle between Perspective and Orthographic Views
Numpad 2, 4, 6, 8 (arrow keys)	Move around in the workspace
Numpad +	Zoom in
Numpad -	Zoom out
Numpad 0	Centered view of the selected object



Figure 4. The buttons panel

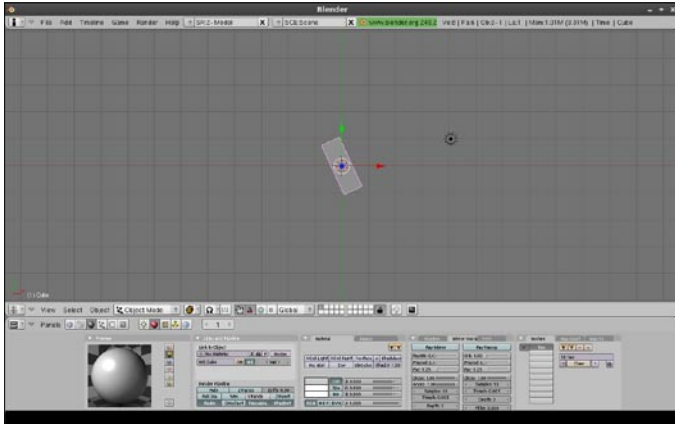


Figure 5. Working in blender

The window contains seven major heads, namely:

1. *View & Controls*,
2. *Edit Methods*,
3. *Language & Font*,
4. *Themes*,
5. *Auto Save*,
6. *System & OpenGL*,
7. *File Paths*.

The best way to learn the nitty-gritty is to experiment with the settings.

Meshes, Vertices and Lights

Transforming Widgets – Object Mode

Before plunging into complex shapes, we need to master the creation and movement of basic meshes. The creation and movement of meshes and most other objects remains similar to what it used to be in almost all earlier versions of Blender (it does not need to change either).

In Object Mode, the main shortcuts used are:

- | | |
|-------|----------------------|
| G key | Move/grab an object |
| S key | Size/scale an object |
| R key | Rotate an object |

A fairly recent addition to Blender is the *Transform Widgets Menu*. Under this, instead of typing the shortcut keys to work with objects, you can simply turn on the widget feature and grab the axis you intend to change. See figure 1 for a snapshot of the menu (Figure 7).

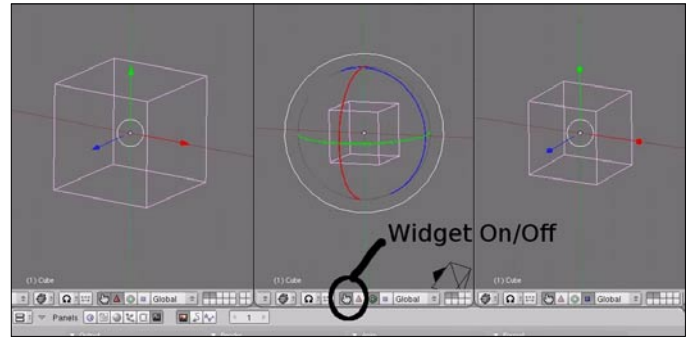


Figure 7. The transform widgets menu

Mesh Vertex Editing – Edit Mode

In any 3D software, mesh and vertex creation is one of the most frequently accomplished tasks. Let's look at the recent innovations in the latest versions of Blender as regards mesh and vertices.

In Blender 2.49b, after creating a mesh, we can go straight into Edit Mode to edit its vertices. In Edit Mode, selected vertices are highlighted in yellow dots while unselected ones are shown in pink dots. In order to select a vertex, you need to right-click on it.

Every object created in Blender 2.49b bears a small dot (generally in its center) which is called the *Object's Center*. Since the center does not always move under Edit Mode, it is advisable to switch to Object Mode before moving objects. If you need to relocate an object's center, simply press *Center Cursor* under Edit buttons (Figure 8).

Viewport Shading

In the recent versions of Blender such as 2.49b and 2.50 Alpha, the Viewport is set to *Solid* shading by default. However, only visible vertices can be selected in *Solid* shading. To switch to *Wireframe* mode, where all vertices can be selected, press the Z key.

Proportional Vertex Editing

Proportional vertex editing is mainly employed in order to create a flow in the shapes when working with vertices. It works only in Edit Mode and the keyboard shortcut is the O key. Proportional vertex editing is dominantly used in the creation of items such as grounds and bevels in 3D scenes. As you progress, the feature you'd be using the most shall be the Knife tool. Notwithstanding that, we can



Figure 6. The user preferences window

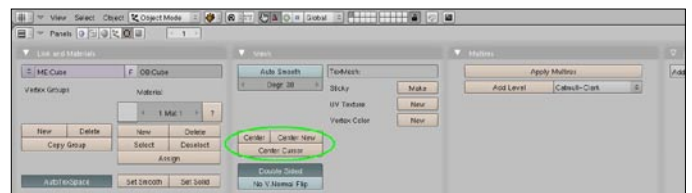


Figure 8. The center cursor button

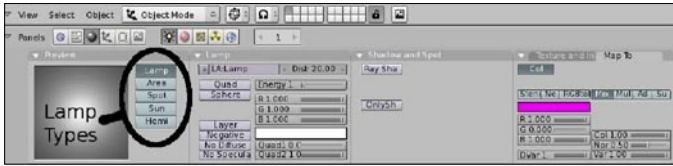


Figure 9. Different types of lamps and lights

Table 2. Keyboard shortcuts for working with objects

Lamp-	Basic Blender lamp which shines in all directions.
Area-	Provides large area lighting and can be scaled.
Spot-	Shines a direct angle of light.
Sun-	Provides an even angle of light, regardless of placement from objects.
Hemi-	A wider light.

safely bypass proportional vertex editing in this article as the methods and techniques employed have remained unchanged since the past couple of years.

Lighting and Cameras

At the most basic level, your work in Blender will not have items that involve the use of a lamp, but will surely have usage for camera. Ideally, even the most minimal scene must have at least 3 or 4 lamps for proper rendering. The major types of lamps or lights used in Blender 2.49b are: see Table 2 and Figure 9.

There have slight alterations in the mode of lamp creation in Blender. To create a lamp in the present version, place the 3D cursor at the desired location and press **SPACE** and select **Lamp->Type**. You will see various options associated with lamps as shown in Figure 3. The best way to implement lamps fully is to experiment with the options and tweak your way through things (after all, where is the fun in going by orthodox style tutorials)!

As regards cameras, your scene is expected to have one by default and it should suffice unless you intend to do something outwardly (such as creating a 3D Jackie Chan stunt simulation). However, if you do plan to have more cameras, simply use the Space Bar. To toggle between active cameras, press **Ctrl** and Numpad 0. The

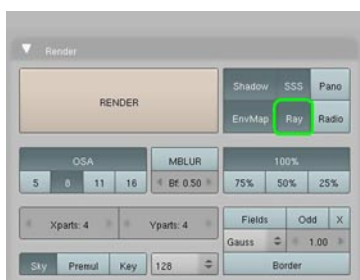


Figure 10. Ray options in blender



Figure 11. The modifier tools' panel

most recent innovation in Blender 2.5 seems to be the addition of tweakable lens length which you can set up as you would in a real camera. Personally, I retain a 35mm length for most of my works.

Raytracing, Text, Movie and Game Engines

Ray Trace Your Shadows!

Raytracing is used to create mirrored and reflective surfaces or to cast object shadows and transparency. It is advisable to use it judiciously as heavy raytracing tends to intensify render times. In Blender 2.49b, to get raytracing to work, you will need to go to the **Render Buttons** menu and turn on **Ray** (see Figure 10). However, unless you are doing something as grand as animations for television, raytracing won't be of much use to you.

Working With 3D Text

Creating, editing and modifying 3D objects and scenes has been covered in detail in previous editions of LinuxForYou. Its time to play around with 3D text.

To create text in Blender, choose the desired location, hit **SPACEBAR->ADD->TEXT**, and a sample text should appear. Modify it as you wish, and then hit **TAB** to exit. Text based commands are found in Edit Buttons, as mentioned in Part I of this tutorial. For instance, to add text on curve, first place a curve using **SPACEBAR->ADD->CURVE** and then use Edit Buttons to insert **Text on Curve**.

Tip

Blender 2.49a and later versions have a keyboard shortcut of **Alt+C** to convert 3D text into a mesh or curve.

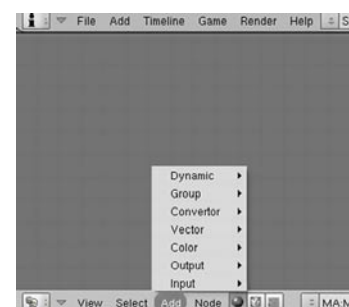


Figure 12. The nodes' menu

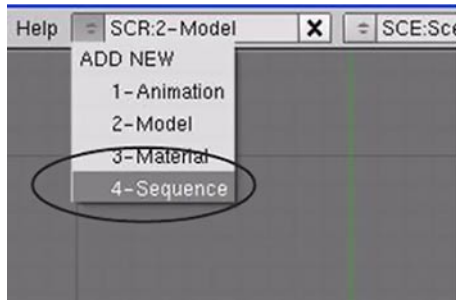


Figure 13. The sequence option

Modifiers and Nodes

In version 2.49b, the location of Modifiers has been altered to place them in a similar and more feasible location. To add a modifier, select the element you wish to add modifier to, and then on the Edit Buttons Menu, under MODIFIERS PANEL, click on ADD MODIFIER (see Figure 11).

Nodes, the most recent addition to Blender, are useful for rendering and post production measures. You can consider nodes to be modules or templates, the difference being that they are less user-defined. The implementation of nodes changes quite quickly, so the best bet is to keep an eye on the Blender.org Wiki. Since Blender 2.49a, the latest nodes are: see Table 3 and Figure 12.

That sums up the summary of new Blender features in recent years. Now, let's get to the business end of things (evil grin). Blender has newly incorporated two terrific concepts, the first one being the ability to create MPEG movies.

Creating a Movie

Technically speaking, a movie is a conglomerate of short clips or images combined together with sounds and effects. Yes, Blender can help you build that conglomerate.

Blender 2.49b comes with a preset screen for sequence editing. To access it, click on *4-Sequence* option in the top toolbar (see Figure 13).

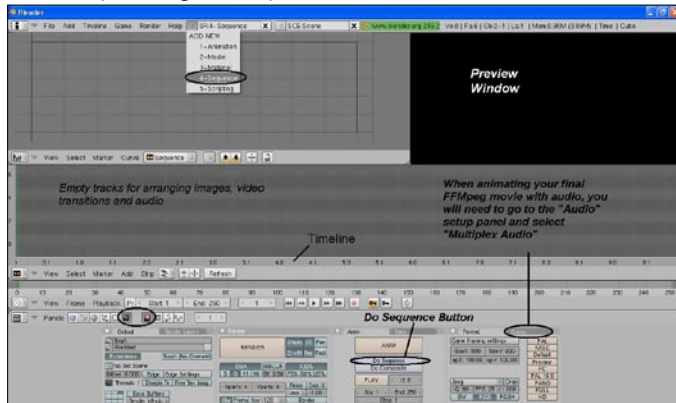


Figure 14. Creating a movie in blender

Table 3. Major lamps/lights available in blender

Group	To specify user-defined blocks of nodes
Distort	To change the shape of the image
Matte	To mask off image areas
Convertor	To change formats and/or separate colors
Filter	To enhance or blur images
Vector	To change or intensify reflections
Color	Color, brightness, contrast, transparency settings
Output	To preview the results
Input	To add an image on the Node Map

Doing that would make your screen look something like Figure 14. Do not panic!

Now, creation of a video basically requires some moving around to do (metaphorically). First, set up the options in Render Buttons, and then press the *DO-Sequence* button. Next, press the *Add* button above the Buttons' window to insert images, audio and movie effects. Insertion of images is simple but while inserting audio files, be aware that not all formats offer equally good performance. As a general convention, formats like WMA should be avoided because more often than not they are finicky. I prefer using WAV, but be warned that it considerably increases the size of the output file. For general movie making purposes, the effect you should be concerned about is Crossfade.

This is it! You are good to go with your movie. Preview it, save it or discard it! The choice is yours.

Basics of Game Engine

One of the most prominent plus point of Blender that helps it to stand apart from the crowd is its Game Engine (known to the geek community as *Real Time Animation Features*). The engine combines physics and logic blocks with animation. You can add/lessen gravity, specify force and friction, etc. In addition, though some level of programming skills in Python are wonderful, they are not necessarily required to work with the Game Engine.

Before going any further, you need to set up the Game Engine. Navigate to the Shading and World Buttons. Under *Mist/Stars/Physics* tab, set the engine to *Bullet*. You may specify the *Gravity* at this junction, though more often than not the default settings should suffice.

FIGURE 15.JPG COMES HERE

Next, move the cursor into the 3D window and press *P*. Click the *Add* button under *Sensors, Controllers and Actuators*. Once you change the sensor from Always to Keyboard, you will see a block for Key. Click in that box and type the key you want to use. For instance, you can tie a force to the Up Arrow, so that when pressed, the sphere moves forward.



Figure 15. Game engine options

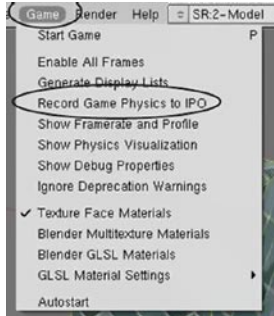


Figure 16. The game menu

The next step is to deal with Physics. Just head to the Game pull-down menu and select *Record Game Physics to IPO* option. Hit *P* to run the action, and use *Esc* to

exit. The action will be written and will henceforth run via *Alt+A*.

Tip

Remember to turn off the *Record Game Physics to IPO* button else it will make a new curve everytime you hit Play (Figure 16).

Well, that sums up this short voyage we embarked on to cover the recent innovations in Blender. Hope you enjoyed the description of the Open Source wonder named Blender! Do write in with your experiences/experiments!

SUFYAN

Sufyan is a 20-year old freelance writer, graphic artist, programmer and photographer based in India. He writes for several print magazines as well as technology blogs. He is also the Founder and Editor-in-Chief at <http://www.bravenewworld.co.nr> He can be reached at <http://www.sufyan.co.nr>

a d v e r t i s e m e n t

RootBSD

PREMIERE VPS HOSTING

Latest FreeBSD

Full Root Access

Starting at \$20/mo

VPS and Dedicated

Multiple Datacenter Locations

Friendly, Knowledgeable Support Staff

WWW.ROOTBSD.NET

Useful OpenBSD Tools

Generally speaking the UNIX world is famous for the rich set of tools it provides and the way it integrates with the rest of the system.

If the tools individually could not perform anything great they become very powerful with the age old UNIX techniques of piping, redirection and backgrounding. There are several other features offered by shells.

Normally we find that most do not depend upon the shell giving a certain feature.

The tools directly use the OS level functions such as signals or background processing.

So this gives us multiple ways to achieve our goal with a particular tool.

The variety and creativity offered by UNIX tool set is mind boggling.

Sometimes one can get overwhelmed by the rich literature in man pages and the features a tool offers.

The fact that most of them can be effectively used in a batch mode with simple text mode commands make them even more tenable to straight forward use with some commercial application or pet project.

In this article I will demonstrate certain tools in the BSD world particularly, OpenBSD, that I use frequently.

- 1) dump(8) and restore(8)
- 2) qemu
- 3) sha1
- 4) ifconfig
- 5) relayd
- 6) spamd

Some of these are not really tools but daemons or programs that come with the base OS. Which is to say that every installed OpenBSD system would have these available.

In fact except qemu, all the tools are available without any extra package being added. Let us now look at one after another in turn.

dump(8) and restore(8)

The tools dump and restore are used for backing up and restoring a partition. The 8 in brackets signify that the tools are administrative in nature.

Being English words themselves the tools are normally referred to in this fashion to avoid confusion.

dump(8) is used to take a binary dump of the filesystem data. It is filesystem specific which means that you can restore them on a different machine of a different architecture. And as opposed to dd(1), it would not copy all the raw disk blocks. In this respect dump(8) is intelligent and also a somewhat slower since it does a great deal more work than dd.

restore(8) is used to do the reverse of dump(8). You can completely image a partition in total like this.

```
# newfs /dev/sd0a
# mount /dev/sd0a /target
# cd /target
# dump af - /dev/wd0a | restore rf -
```

Please be very careful.

All these commands are to be run as root. And I am cloning the filesystem data from the disk wd0 partition a to disk sd0 partition a. dump(8) normally writes to a binary file.

```
# dump af foo.bin /dev/wd0d
```

would create a single file foo.bin with all the contents of the /dev/wd0d partition.

Only the parts that have allocated disk blocks are written, not the entire filesystem space.

And restore also operates on a file like this. You can copy this file to a remote machine using ssh or ftp then run this command.


```
# restore rf foo.bin
```

But before `restore(8)` you must to format the filesystem with `newfs(8)` and mount it.

Before formatting you would need to create the partition using `disklabel(8)`. Now that brings us to the next tool I like. `qemu`.

Qemu

Qemu is a 100% open source implementation of emulation which doubles up as virtualization or cloud as people like to call it.

It is particularly important to me since I am an appliance guy and I have many products in the networking appliance marketplace.

And I cannot survive physical reboots and ISO burns and hard disk formatting just to test my code.

Instead I simply use `qemu` which allows me to run my OS just like I would run any application. The great thing is that I can `dd(1)` a USB stick to a single file and start up with `qemu` and it just works!

It is quite amazing since `qemu` supports user mode networking which allows you to use any TCP service running outside like mail, ftp or http, while preventing access to access any TCP or UDP or ICMP running inside the `qemu` guest.

This is done by using `qemu` in bridge mode. That would exactly be like connecting an additional physical machine to your switch.

I have a VPN product and `qemu` allows far easier testing than would otherwise be available. You simply run `qemu` like this at a very basic level.

```
$ qemu -cdrom foo.iso
```

If `foo.iso` is a liveCD. You can test LiveCDs without wasting optical media. And `qemu` also has the ability to use the host machine's audio ports.

It is fast, convenient and fun. But it has a steep learning curve. In my case it took around 2 years and even now there are many things I don't know.

sha1

This is a really simple tool. I use this for integrity checks. Just run it like this.

```
$ sha1 /etc/passwd
SHA1 (/etc/passwd) = bfe2be6875743ea537ca24604662b9684bbdcf5f
```

It produces a fixed size output which is the a hash of the original file.

Once you download an ISO or a binary image you can run `sha1` on both sides and verify the integrity. It performs a single function but very useful.

ifconfig(8)

`ifconfig` is a command that everyone knows as it is used for configuring a network interface of a machine.

I like it because under OpenBSD `ifconfig` is also used for creating bridges and nearly everything related with networking. I can use `ifconfig` like this to create `bridge(4)` ports or even `trunk(4)`.

```
# ifconfig bridge0 create
# ifconfig bridge0 add em0 add em1 up
```

This would create a bridge with two interfaces `em0` and `em1` as part of it. Really simple.

Contrast this with Linux. You need to install a package for it. `trunk(4)` is an interface type created by Reyk Floeter to solve some problem he had long ago.

It allows interface level failover and load balancing. You can create a trunk port to failover between a wired and a wireless network simply like this.

```
# ifconfig trunk0 create
# ifconfig trunk0 trunkproto failover trunkport bge0
                                trunkport em0 192.168.1.10
                                netmask 0xffffffff0
```

`ifconfig(8)` can do a lot more particularly for wireless networks. But I have not yet played with them since I don't have a laptop. You can create IP aliases with this command.

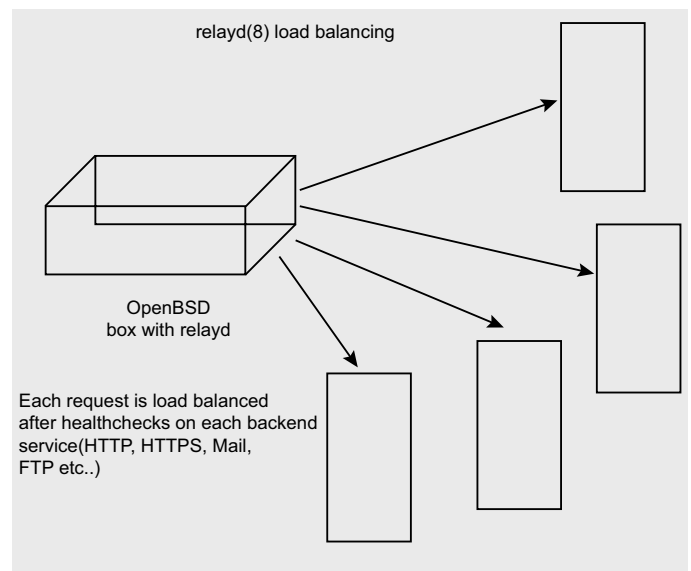


Figure 1. Relayd load balancing

```
# ifconfig r10 alias 172.16.1.1
```

You can create any number of aliases on an interface and this is a powerful tool for doing advanced networking tricks. In addition to `ifconfig`, `netstart(8)` is useful.

```
# sh /etc/netstart
```

works mostly correctly when you have setup networking correctly. This would simulate a for the network interface. There are many situations under which this does not help but mostly it helps sort out networking problems without requiring a reboot.

relayd(8)

`relayd(8)` is a failover and load balancing daemon which does what is known as service redirection based on health checks of applications. It is also developed by Reyk but it works at a much higher level. It can be used for very sophisticated layer 7 filtering, on the fly rewriting and so on.

Proxying, load balancing implementing direct server return and so on.

You interact with the daemon using `relayctl` which internally uses a UNIX domain socket.

Here is a simple example to do very basic level failover between hosts.

```
host1="192.168.1.2"
host2="192.168.1.3"
table <hosts> {
    $www1
    $www2
}

table <cvs> { 192.168.1.1, 192.168.1.2, 192.168.2.3 }

redirect „www“ {
    listen on www.foo.com port 80
    forward to <cvs> check http „/" code
    200
}
```

Refer to the manpage for details. You can do SSL acceleration and HTTP session persistence with it. I did not yet get an opportunity to play with it yet. So my knowledge is quite limited.

spamd(8)

`spamd(8)` along with `spamlogd(8)` and `spamd-setup(8)` is useful for spam control. It is used by sites running mail servers to protect against the botnet style spam.

It works remarkably well for nearly every class of spam but then there are limitations. It does not provide content scanning or virus filtering. It is too confusing.

For instance `spamd` is a fake SMTP daemon that acts as a tarpit that forces mail senders to be standards compliant. A great deal of real world servers are not and that means that certain changes need to be made to it to adapt with the evolving needs of the marketplace.

In fact `spamd(8)` supports multicast and unicast synchronization between multiple hosts running the spam control daemon.

A simple `pf(4)` rule like this can enable spam protection.

```
pass in on r10 proto tcp from any to any port smtp \
    rdr-to 127.0.0.1 port spamd
pass in on r10 proto tcp from any to any port smtp rdr-to
    192.168.1.20
```

This of course assumes that we run the mail server on a different machine. This will certainly also work with a locally running mail server as long as you change the rule appropriately.

The main attraction of `spamd(8)` for me is that it saves precious bandwidth and it is a network level spam filter. It is mail server agnostic which is really nice.

That brings us to the topic of mail servers and Gilles is busy developing OpenSMTPD. It will take some more time before we hopefully get to see world's best SMTP implementation.

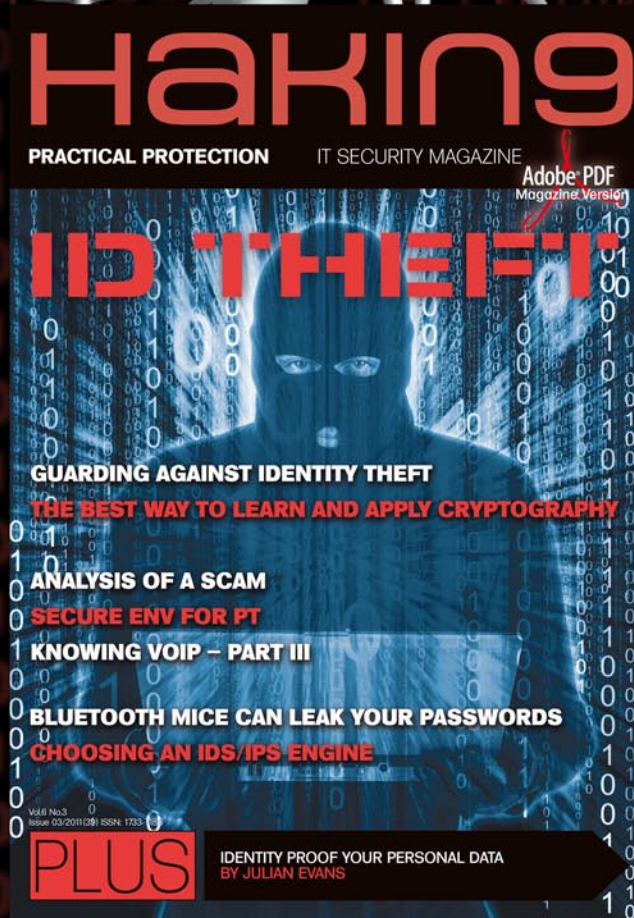
Have fun with OpenBSD.

GIRISH VENKATACHALAM

Girish has close to 15 years of UNIX experience and he enjoys OpenBSD more than anything else in the technology world.

HAKING

PRACTICAL PROTECTION



IT SECURITY MAGAZINE

MAGAZINE

BSD

In the next issue:

- Benchmarking Different Kind Of Storages**
- Rump anykernel architecture for NetBSD**
- and Other !**

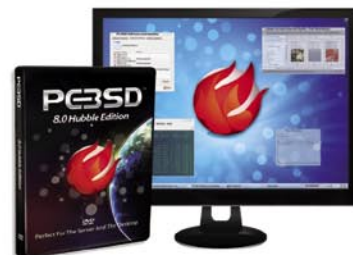
**Next issue is coming in
April!**



FreeBSD Mall

Your FreeBSD & PC-BSD Resource

www.FreeBSDMall.com



FreeBSD 8.1 Jewel Case CD/DVD

Set contains:

- Disc 1: Installation Boot (i386)
- Disc 2: LiveFS (i386)
- Disc 3: Essential Packages (i386)
- Disc 4: Essential Packages (i386)

FreeBSD 8.1 CD	\$39.95
FreeBSD 8.1 DVD	\$39.95
FreeBSD 7.3 CDROM	\$39.95
FreeBSD 7.3 DVD	\$39.95

FreeBSD Subscriptions

Save time and \$\$\$ by subscribing to regular updates of FreeBSD!

FreeBSD Subscription, start with CD 8.1	\$29.95
FreeBSD Subscription, start with DVD 8.1	\$29.95
FreeBSD Subscription, CD 7.3	\$29.95
FreeBSD Subscription, DVD 7.3	\$29.95

PC-BSD 8 DVD (Hubble Edition)

PC-BSD 8 DVD	\$29.95
PC-BSD Subscription	\$19.95

BSD Magazine

BSD Magazine	\$11.99
BSD Magazine Subscription	\$11.99

The FreeBSD Handbook

The FreeBSD Handbook, Volume 1 (User Guide)	\$39.95
The FreeBSD Handbook, Volume 2 (Admin Guide)	\$39.95
★ Special: The FreeBSD Handbook, Volume 2 (Both Volumes)	\$59.95
★ Special: The FreeBSD Handbook, Both Volumes, & FreeBSD 8.1	\$79.95

The FreeBSD Bundle

Inside the Bundle, you'll find:

- FreeBSD Handbook, 3rd Edition, Users Guide
- FreeBSD Handbook, 3rd Edition, Admin Guide
- FreeBSD 8.1 4-disc set
- FreeBSD Toolkit DVD

★ Special: The FreeBSD CD Bundle	\$89.95
★ Special: The FreeBSD DVD Bundle	\$89.95

The FreeBSD Toolkit DVD

FreeBSD Mousepad

FreeBSD Caps

PC-BSD Caps

For **MORE** FreeBSD & PC-BSD items, visit our website at **FreeBSDMall.com!**

CALL 925.240.6652 Ask about our software bundles!

t-shirts
\$18-\$21.99



Enterprise Open Source Storage: FreeNAS with ZFS has arrived...

✓ Reliable ↑ Scalable \$ Affordable



1-855-GREP-4-IX | <http://www.iXsystems.com/FreeNAS>

